Fragments structuraux : comparaison, prédictibilité à partir de la séquence et application à l'identification de protéines de virus

Clovis Galiez

ANNÉE 2015

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

École doctorale Matisse

présentée par

Clovis GALIEZ

préparée à l'unité de recherche IRISA – UMR6074
Institut de Recherche en Informatique et Système Aléatoires

Fragments structuraux : comparaison, prédictibilité à partir de la séquence et application à l'identification de protéines de virus

Thèse soutenue à Rennes le 8-12-15

devant le jury composé de :

Rapporteur ANNE-CLAUDE CAMPROUX
Professeure à l'Univeristé Paris-Diderot / *Rapporteur*

Rapporteur DAVE RITCHIE
Directeur de recherche chez Inria Nancy / *Rapporteur*

Examinateur RUMEN ANDONOV
Professeur à l'Université de Rennes 1 / *Examinateur*

Examinateur MARC SEBBAN
Professeur à l'Université de Saint-Étienne / *Examinateur*

Directeur FRANÇOIS COSTE
Chargé de recherche chez Inria de Rennes / *Directeur de thèse*

Directeur JACQUES NICOLAS
Directeur de recherche chez Inria Rennes / *Co-directeur de thèse*

Table des matières

Introduction

Les protéines sont des macro-molécules qui jouent un rôle clé dans la plupart des processus biologiques. Leurs fonctions vont du transport de molécules (telle l'hémoglobine pour le transport l'oxygène dans le sang) à la catalyse de réactions chimiques (comme les enzymes de digestion) en passant par le transport de signaux (l'insuline informe par exemple les cellules du taux de sucre) ou la régulation de l'expression des gènes avec les facteurs de transcription. Leur étude permet de mieux comprendre les mécanismes sous-jacents à des phénomènes biologiques comme le fonctionnement métabolique d'un organisme ou l'adaptation d'une espèce à un environnement, mais aussi plus concrètement de pouvoir cerner les causes de pathologies.

Ces dernières décennies, l'avancement de la technologie ne cesse de faire croître la quantité de données biologiques disponible numériquement. Cette quantité d'information ne peut être traitée sans l'aide de la *bioinformatique* qui s'attache à corriger, filtrer, classer et analyser ces données. En particulier, l'*identification des protéines et de leur fonction* est une étape clé de plus en plus sollicitée par la production croissante de données. Cette thèse s'inscrit dans cette problématique d'identification en formalisant quelques concepts menant à la définition de nouveaux types de signatures protéiques.

On peut voir métaphoriquement une protéine comme un enfilement de perles magnétiques : les perles vont s'attirer et se repousser dans les 3 dimensions jusqu'à aboutir à un état stable. Plus précisément, les protéines sont composées d'acides-aminés liés entre eux par des liaisons peptidiques formant une chaîne. Dans le vivant, il existe 20 types différents d'acides-aminés ayant des propriétés physico-chimiques différentes (taille, polarité, capacité d'interaction, etc.). L'enchaînement de ces différents acides-aminés le long de la chaîne peptidique formant la protéine est appelé la *séquence* ou la *structure primaire* de la protéine. La conformation spatiale adoptée par la chaîne peptidique qui est déterminée par un minimum d'énergie interne est aussi appelée *structure tertiaire* ou plus simplement *structure* de la protéine. En 1972, un tournant est marqué par la publication de l'expérience d'Anfinsen [Anf72] qui montre que pour une même séquence, la conformation atteinte par le minimum d'énergie interne à température ambiante est toujours la même. On réalise ainsi que la conformation d'une protéine est entièrement déterminée par la séquence d'acides-aminés la composant. On résume souvent ce résultat en disant que *la structure primaire détermine la structure tertiaire.*

L'agencement spatial des acides-aminés ainsi déterminé par le minium d'énergie interne conditionne la capacité d'une protéine à se lier à d'autres composés chimiques ou biologiques et détermine donc la fonction de la protéine (catalyse d'une réaction

chimique spécifique, formation d'un compartiment par liaison à d'autres protéines, régulation de la transcription d'un gène par liaison à l'ADN, etc.). Le fait que la séquence détermine la structure qui elle-même détermine la fonction de la protéine est appelé le *paradigme séquence-structure-fonction.*

Un moyen de déterminer la fonction d'une protéine consiste alors à comparer sa structure ou sa séquence à des protéines de fonction connue [SW10] : la fonction aura d'autant plus de probabilité d'être identique que la séquence ou la structure sera similaire.

Les séquences nucléiques ainsi que les séquences protéiques associées évoluent au fil des générations par des mécanismes de mutations (substitutions, insertions et délétions), donnant naissance à des protéines de structure et éventuellement de fonction différente. Ces fonctions subissent alors une pression exercée par l'environnement : les structures fournissant une fonction vitale ou un avantage compétitif seront conservées.

Les structures changent cependant moins rapidement que les séquences sous-jacentes car il faut en général plus de 60% de différence entre deux séquences pour avoir une différence significative de structure. On dit généralement que *la structure est plus conservée que la séquence*. Réciproquement, pour deux protéines de structure similaire – et donc de fonction identique – il n'est pas toujours possible de détecter une similarité de séquence. En effet, les séquences peuvent avoir tant divergé et qu'il devient difficile de détecter un apparentement. A l'inverse, il se peut peut que deux séquences d'origine entièrement différente aient évolué sous la pression de sélection pour converger vers une même structure sans partager aucune similarité de séquence (on parle dans ce cas d'*homoplasie*). Comme il peut exister une large diversité de séquence pour une même structure il est donc plus direct d'utiliser l'information structurale afin de déterminer la fonction.

Bien souvent, la conservation est effective seulement sur quelques portions de protéines qui sont nécessaires à la fonction. Ainsi, on peut observer la présence de portions de séquences et de structures conservées à travers des lignées d'organismes. Cette conservation à l'échelle locale permet d'identifier la fonction d'une protéine : on établit des signatures qui identifient des portions de séquence ou de structure comme caractéristiques d'une fonction. Par exemple, un site catalytique est généralement composé de quelques acides-aminés fortement conservés avec des caractéristiques structurales spécifiques au type de ligand auquel la protéine doit se lier. Un tel motif structural constitue une signature : dès lors qu'il est identifié dans une protéine on peut présumer de sa fonction.

Pour pouvoir déterminer la structure d'une protéine, il faut dans un premier temps réussir à l'isoler (étape de purification) du reste des composés biologiques et chimiques présents dans son environnement naturel. Une fois purifiée il existe ensuite deux principales méthodes : la diffraction de rayons X par des cristaux protéiques et la résonance magnétique nucléaire (RMN). Chacune de ces techniques possèdent ses avantages et ses limites, et si elles permettent de résoudre la structure des protéines à l'échelle atomique, cela reste long et coûteux à mettre en œuvre.

Par ailleurs, la séquence d'une protéine est la traduction d'une séquence nucléique contenue dans le génome de l'organisme qui l'exprime. Grâce aux technologies de sé-

quençage de l'ADN et aux techniques de prédiction de gènes, on peut avoir accès aux séquences de l'ensemble des protéines d'un organisme. Cela permet d'identifier des protéines sans nécessiter de les observer, ni de les isoler. En conséquence, la séquence des protéines est davantage disponible que leur structure. Il arrive donc souvent qu'il soit nécessaire de déterminer la fonction associée à une séquence observée dans un génome, sans que la structure soit connue ni même que la protéine n'ait pu être mise en évidence expérimentalement. Par exemple en 2015, 112 722 entrées sont disponibles dans la principale base de données de structures PDB [BKW+77] alors que la base de données UniprotKB/TrEMBL [MC11] référence 51 374 999 séquences protéiques, soit un peu moins de 50 fois plus.

Si nous voulons combiner les avantages de fiabilité de la signature structurale et la disponibilité des séquences, une possibilité est de construire des signatures de séquences pour lesquelles la structure associée est une signature structurale. Dans un tel cas, on voit qu'on cherche à obtenir des signatures structurales détectables à partir de la séquences.

Par exemple, une signature structurale peut prendre la forme d'une structure fortement conservée et caractéristique de la fonction. La signature de séquence associée devra alors correspondre uniquement à cette structure. Nous formaliserons cette idée par la notion de structure *prédictible* à partir de sa séquence.

La prédictibilité prend davantage son sens à l'échelle locale. En effet, bien que la séquence d'une protéine entière détermine généralement une unique structure, la variabilité est beaucoup plus importante à l'échelle locale : un segment de séquence peut se traduire par différentes structures d'une protéine à l'autre. En effet, l'environnement alentour exerce des interactions électro-magnétiques et influence la structure du fragment. On peut alors se demander quelles sont les conditions pour qu'au niveau local la séquence détermine la structure : quelle information sur l'environnement est nécessaire ? peut-on former une librairie de séquences dont la structure est fixe ? une telle librairie peut-elle être construite de manière minimale ? peut-elle couvrir l'ensemble des séquences connues ?

Au travers de cette thèse nous avons amorcé une réflexion vers ces questions. Dans l'optique de définir des structures prédictibles, l'environnement spatial nous paraissait nécessaire à prendre en compte, nous amenant à supposer que la *localité spatiale renforce le caractère prédictible d'une structure*. Ainsi, la notion de contact – *i.e.* les paires d'acides-aminés proches dans une structure – devrait jouer une rôle prépondérant pour définir des structures prédictibles. Par ailleurs, il a déjà été montré que le voisinage séquentiel d'un contact joue également un rôle important pour la prédiction de contact depuis la séquence, ce qui nous a amené à formuler une seconde hypothèse : *le voisinage séquentiel d'un contact renforce le caractère prédictible d'une structure*. Avec ces deux hypothèses nous avons défini les fragments en contact (ou CF pour *contact fragments*) comme des portions de structure permettant de concilier localité spatiale et voisinage séquentiel. Par une simple expérience sur des structures représentatives de l'ensemble des protéines, nous avons pu mesurer le gain de prédictibilité dont les CF bénéficient par rapport aux fragments contigus en séquence ainsi qu'aux paires de fragments qui ne sont pas en contact en structure.

La définition des CF utilise des seuils de distance, ce qui implique que pour deux protéines similaires, les CF similaires peuvent différer à leurs extrémités de quelques acides-aminés. Ainsi, le premier acide-aminé d'un CF dans une protéine peut correspondre au deuxième ou troisième d'un CF d'une deuxième protéine. Afin de pouvoir comparer les CF entre eux, les scores de similarité structurale actuels nécessitent alors de passer par une phase d'alignement permettant d'établir une correspondance entre les acides-aminés à comparer. Cette phase d'alignement n'est cependant pas toujours adéquate suivant les applications. En effet, d'une part les parties non-alignées sont évincées du score, ce qui peut nuire à la comparaison globale de la structure, et d'autre part ces comparaisons ne respecteront par l'inégalité triangulaire qui est une propriété importante pour pouvoir former des groupes de structures similaires. C'est pourquoi nous avons introduit l'ASD (pour Amplitude Spectrum Distance) qui utilise la transformée de Fourier de la matrice formée par les distances inter-atomiques pour comparer la forme *globale* des CF. L'ASD, qui ne nécessite pas d'alignement préalable des acides-aminés, respecte l'inégalité triangulaire tout en étant tolérante à un décalage d'acides-aminés ainsi qu'à d'éventuelles insertions ou délétions. Au cours de notre investigation de cette mesure de dissimilarité structurale, nous avons remarqué que ses propriétés étaient non seulement valables pour la comparaison de CF, mais que l'ASD pouvait aussi s'appliquer universellement à la comparaison de fragments simples de structure (*i.e.* de polypeptides). Nous avons pu démontrer certaines propriétés théoriques de l'ASD et montrer qu'elle offrait de meilleures performances que les scores existants sur des tâches concrètes de classification de fragments et de fouille structurale.

La définition des CF ainsi que la possibilité de les comparer et de les classifier de manière fiable avec l'ASD permet de définir des signatures structurales. On peut voir les CF comme des descripteurs de structure et chercher à déterminer les CF qui permettent de caractériser une famille de protéines. L'identification des CF caractéristiques d'une famille dans une protéine inconnue permet alors de supposer que celle-ci pourrait être apparentée à la famille. De plus, d'une manière générale, comme les CF bénéficient d'une bonne prédictibilité ils permettent alors d'obtenir des signatures qu'on pourrait qualifier de *structuro-séquentielles*.

Nous avons utilisé concrètement les CF pour l'apprentissage de signatures caractéristiques de séquences de protéines virales. Cette problématique a initialement émergé lors d'une collaboration [BCC+14] au projet PEPS VAG qui avait pour but d'identifier les virus dans les séquences d'ADN viral provenant de l'expédition Tara Oceans. Les virus sont des entités biologiques extrêmement importantes : ils peuvent être la cause de pathologies graves animales comme humaines, mais ils sont aussi fortement impliqués dans des phénomènes d'échelle globale tels que l'absorption du CO_2 par les océans [LKS+14]. Pour autant, les protéines virales sont bien souvent mal caractérisées tant au niveau de la structure que de la séquence. Après les premières études [BCC+14] réalisées dans le cadre du projet VAG, au cours d'un séjour de 3 mois à *University of California in Irvine* dans le laboratoire de Pierre Baldi, j'ai pu développer la suite VIRALpro (détaillé en section 7.3) qui permet l'identification de séquences de protéines de capsides (protéines servant à envelopper le matériel génétique du virus) ainsi que l'identification de protéines de queue (qui permettent de caractériser la taxonomie de certains

types de virus). VIRALpro classifie les séquences grâce à des techniques d'apprentissage automatique (SVM et boosting de préférence) qui combinent l'identification depuis la séquence de différentes caractéristiques structurales locales (détection de CF spécifiques au protéines de capside et de queue, prédiction du taux de structure secondaire).

Le prochain paragraphe présente l'organisation de ce manuscrit qui s'articule davantage selon un ordre de dépendance des concepts que selon la chronologie d'apparition des idées telle que nous venons de la présenter.

Organisation du manuscrit

Dans un premier temps nous ferons un bref rappel sur les protéines, leurs descriptions, leur origine et leur évolution (chapitre 1). Nous verrons ensuite quelles sont les techniques classiques pour pouvoir comparer des protéines au niveau de leur séquence ainsi que de leur structure (chapitre 2). Nous décrirons alors quelques méthodes de caractérisation d'ensemble de protéines, ici encore au niveau de la séquence aussi bien que de la structure (chapitre 3).

Nous aborderons ensuite les contributions (partie II) propre à cette thèse en exposant en premier lieu une brève formalisation du lien séquence-structure dans les protéines (chapitre 4). Nous présenterons ensuite l'ASD au chapitre 5 ainsi que quelques expérimentations évaluant sa capacité à comparer des fragments contigus de structure (section 5.5). L'ASD nous permettra alors de comparer les fragments en contacts présentés dans le chapitre suivant (chapitre 6). Nous montrerons alors que les fragments en contact jouissent d'une bonne prédictibilité (section 6.4). Nous verrons enfin quelques applications à des données réelles de protéines structurales de virus (chapitre 7) au niveau de leur caractérisation en structure (section 7.1) ainsi que de leur détection en séquence par apprentissage automatique avec l'outil VIRALpro développé dans cette thèse (section 7.3).

Notations

Nous définissons ici quelques notations que nous utiliserons tout au long de ce document.

Notation	Dénotation
$[\![i, j]\!]$	Intervalle des *nombres entiers* compris entre i et j.
$\binom{N}{k}$	Coefficient binomial k parmi N.
$P(X = x)$	Probabilité que la variable aléatoire X prenne la valeur x.
$\|E\|$	Cardinal de l'ensemble E
$\|x\|$	Module du nombre complexe x : si $x = A.e^{i\phi}$ avec $A, \phi \in \mathbb{R}$, $\|x\| = A$
$\arg x$	Argument du nombre complexe x : si $x = A.e^{i\phi}$ avec $A, \phi \in \mathbb{R}$, $\arg x = \phi$
$dom(T)$	Domaine (ensemble) de définition de la fonction T.
$img(T)$	Image (ensemble) $T(dom(T))$ de la fonction T.
X^t	Transposée de la matrice X.
$\det X$	Déterminant de la matrice carrée X.
$\mathbf{1}_B(x)$	Fonction caractéristique du prédicat booléen B : $\mathbf{1}_B(x) = \begin{cases} 1 \text{ si } B(x) \text{ est vrai} \\ 0 \text{ sinon} \end{cases}$

Première partie

Etat de l'art

Chapitre 1

Les protéines : séquence, structure, fonction

Dans ce chapitre nous introduisons les éléments sous-jacents à la séquence et la structure des protéines. Nous décrirons alors brièvement les mécanismes évolutifs qui, en sélectionnant les avantages compétitifs sur les fonctions des protéines, régissent la conservation de la séquence et de la structure.

1.1 Différentes échelles de description

Nous présentons ici les différentes échelles de description des protéines depuis le niveau de l'acide-aminé (diamètre de l'ordre de quelques Angströms) jusqu'au niveau global de leur structure (d'un volume de l'ordre de la dizaine de nm^3).

Le lecteur désirant approfondir les quelques notions de base exposées ci-après peut se référer à un ouvrage de biologie structurale tel que [AL09].

Acide-aminé Un acide-aminé est un composé organique pouvant former une chaîne polymérique. Il est composé (voir figure 1.1) d'un atome de carbone "central" (appelé carbone alpha, noté C_α) lié à un groupe CO, d'un atome d'azote, d'un atome d'hydrogène et enfin d'une chaîne dite "latérale", dénotée génériquement "R" (pour *résidu*) dans la littérature. Dans le domaine du vivant, 19 types de chaînes latérales permettent de différencier chaque acide-aminé (pour un nombre total de 20 acides-aminés, la glycine n'ayant pas de chaîne latérale). Ainsi, suivant le type de chaîne latéral, l'acide-aminé aura des propriétés physico-chimiques différentes. La table suivante illustre les noms des acides-aminés du vivant ainsi que deux de leurs propriétés :

Nom	Symbole	Chargé	Polaire
Ala	A		
Cys	C		X
Asp	D	X	X
Glu	E	X	X
Phe	F		
Gly*	G	-	-
His	H	X	X
Ile	I		
Lys	K	X	X
Leu	L		
Met	M		
Asn	N		X
Pro	P		
Gln	Q		X
Arg	R	X	X
Ser	S		X
Thr	T		X
Val	V		
Trp	W		X
Tyr	Y		X

*Acide-aminé n'ayant pas de chaîne latérale

TAB. 1.1: Quelques propriétés physico-chimiques des acides-aminés.

L'attribut "chargé" signifie que la chaîne latérale possède une charge électrostatique strictement positive ou négative et l'attribut "polaire" signifie que les charges négatives et positives sont réparties de manière hétérogène et forment un dipôle électrostatique. Par exemple un acide-aminé polaire aura tendance à attirer d'autres molécules polaires comme l'eau (on parle alors d'acide-aminé hydrophile).

Il existe bien sûr d'autres propriétés telles que la taille, la présence de cycles aromatiques, la capacité à former des ponts hydrogène, etc. qui confèrent à chaque acide-aminé sa spécificité physico-chimique.

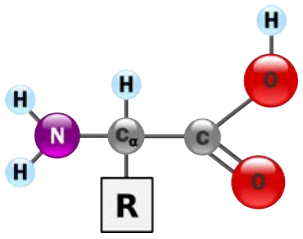

FIG. 1.1: Représentation schématique des atomes constituant un acide-aminé

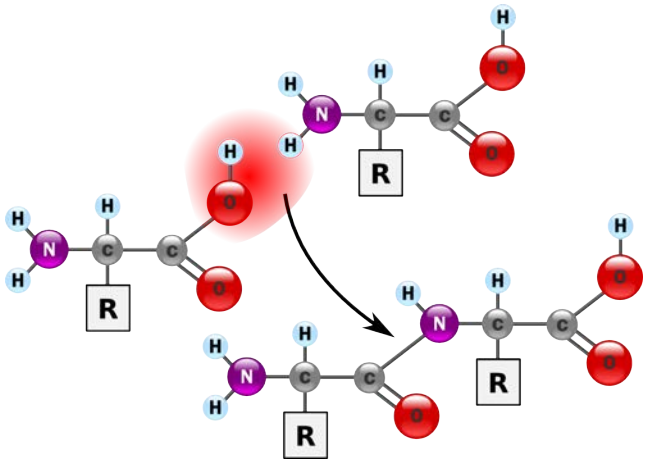

FIG. 1.2: Liaison peptidique

Structure dite "primaire" : le polypeptide La liaison des acides-aminés via le groupe CO et N permet de former une *chaîne polypeptidique*.

Cette chaîne a deux extrémités l'une se terminant par un atome d'azote N (appelée extrémité N-terminale), l'autre par un groupe CO (appelée extrémité C-terminale). Le référencement de la séquence d'acide-aminés en parcourant cette chaîne de l'extrémité N-terminale vers la C-terminale est appelé la *structure primaire* ou *séquence* de la protéine. Formellement, cette séquence peut-être décrite par un mot sur un alphabet Σ de 20 symboles représentant les acides-aminés (cf. tableau 1.1). Dans la suite, on appellera *segment* (respectivement *fragment*), une portion contigüe de séquence (respectivement de structure) du polypeptide.

Il existe de nombreuses bases de données [ABW04] recensant des séquences de protéines trouvées dans la plupart des branches du vivant, dont on peut citer les plus célèbres UniProtKB [MC11] et RefSeq [PTBM12].

La conformation spatiale du polypeptide En 1972, Anfinsen [Anf72] remarqua que lorsqu'on perturbe sa conformation tridimensionnelle, une protéine se replie toujours vers la même forme finale. Il formule l'hypothèse que le minimum d'énergie interne — et donc la conformation spatiale — d'une structure est entièrement déterminé par la séquence des acides-aminés la composant.

En effet, un polypeptide adopte une conformation tri-dimensionnelle déterminée par différentes interactions physico-chimiques comme par exemple des ponts hydrogène et interactions aromatiques (par des acides-aminés polaires), par des interactions de Van Der Waals ou encore par phénomène d'hydrophobicité (concernant les acides-aminés les moins polaires). Bien que cette conformation puisse admettre une certaine flexibilité, et même parfois subisse des changements importants suivant l'environnement (solvant, température), on considère en général uniquement la structure associée à un minimum d'énergie interne dans des conditions physiques standard. Depuis l'expérience d'Anfinsen certaines études montrent que dans tous les domaines du vivant il existe des

protéines ayant des *fragments intrinsèquement désordonnés* [DOM+08] dont la structure ne semble pas avoir de minimum d'énergie stable. Dans le reste de ce document, nous nous intéresserons plus particulièrement aux portions de protéines dont la structure est stable et déterminée par la séquence.

Structures secondaires On remarque que des portions locales de polypeptides adoptent des conformations particulières que l'on retrouve de manière fréquente dans la plupart des protéines. Ces structures particulières sont appelées *structures secondaires* et se déclinent sous deux principales formes : le brin β ayant une structure globale linéaire et l'hélice α ayant un squelette carboné sous forme hélicoïdale d'un pas de 5.4Å avec en moyenne 3.6 résidus par tour d'hélice. Le reste de la structure est souvent dénoté sous le terme de *coil* qui constitue un troisième type de structure secondaire.

On représente de manière simplifiée la structure d'un polypeptide par la courbe décrite par son squelette carboné sur laquelle on schématise par des rubans les deux types de structures secondaires α et β comme dans l'exemple ci-dessous :

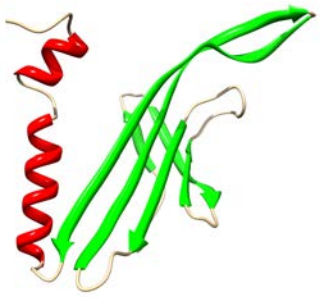

FIG. 1.3: Hélices alpha (en rouge), brins beta (en vert), coudes (en doré). Ref. PDB 2MS2.A.

Superstructure secondaire Les structures secondaires s'organisent aussi fréquemment en structures reconnaissables. Notamment les brins β s'organisent souvent en feuillets dits *feuillets* β, comme dans visible dans la figure 1.3.

Comme la structure de ses superstructures secondaires est plus complexe, la classification est moins évidente et fait l'objet de publications même relativement récentes comme par exemple la classification des différents types de feuillets β en fonction de l'ordre des brins β voisins dans la séquence [CGKG07].

Structure tertiaire On appelle *structure tertiaire* d'une la protéine la conformation tridimensionnelle globale de sa chaîne polypeptidique. La structure tertiaire est généralement déterminée par sa structure primaire, et on peut parfois identifier à une échelle plus fine des sous-parties (les domaines, décrits dans le prochain paragraphe) dont la structure est également déterminée par la séquence.

Domaine protéique Un domaine protéique est généralement défini comme un fragment de protéine dont la conformation tri-dimensionnelle est uniquement déterminée par le domaine-lui même : quelque soit la composition du reste du polypeptide composant la protéine, la structure tri-dimensionnelle du domaine restera fixe à faible variation locale près. La figure 1.4 ci-dessous indique que les domaines — pris sur un sous-ensemble représentatif des structures de protéines connues — ont majoritairement une taille de 100 acide-aminés, et la plupart ont une taille < 200 acides-aminés.

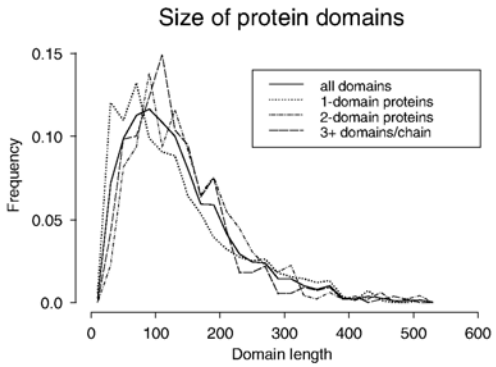

FIG. 1.4: Statistique sur la longueur des domaines d'un sous ensemble représentatif de la PDB. Source [WMBB00].

La conformation tri-dimensionnelle d'un domaine ou plus généralement d'une protéine est parfois dénotée sous le terme *repliement* ou par le mot anglais *fold*. Il existe plusieurs classifications des domaines protéiques, les plus connues étant la classification manuelle SCOP [LCAH$^+$00] et sa nouvelle version SCOP2 [AHC$^+$14], ainsi que la classification automatique CATH [SLC$^+$15]. SCOP est principalement organisé selon la hiérarchie suivante : *classe* (discrimine les structure suivant leur composition en structures secondaires), *fold* (similarité structurale), *super famille* (potentiellement faible similarité de séquence, mais similarité fonctionnelle ou structurale indiquant une origine ancestrale commune probable), *famille* (grande identité de séquence, ou fonction identique avec identité de séquence plus faible). CATH partage le même premier niveau, puis affine la classification suivant les niveaux *Architecture* (organisation spatiale des structures secondaires), *Topologie* (séquence de liaison des différentes structures secondaires), et enfin *Homologie* (similarité de séquence). La classification SCOP2 n'est plus seulement une classification hiérarchique, mais répertorie l'ensemble des structures de protéines sous forme d'un graphe rendant compte simultanément des relations de similarité structurale ainsi que de parenté évolutive.

Les domaines sont souvent vus comme des briques de bases sélectionnées au cours de l'évolution dont l'assemblage permet de composer la fonction de la protéine. Par ailleurs, lorsqu'on s'intéresse à la prédiction de structure à partir de la séquence, on peut se focaliser plus simplement sur la prédiction de domaines, car par définition, la simple connaissance de leur séquence détermine la structure qui leur sera associée.

Ainsi, prédire la structure des domaines à partir de leur séquence constitue une pierre angulaire à la prédiction de structure de protéine.

Structure quaternaire Pour assurer leur fonction, il est parfois nécessaire que plusieurs chaînes polypeptidiques se lient ensemble. On appelle *structure quaternaire* la structure résultante de leurs positions relatives. Leur section est de l'ordre de 1600Å^2 [AL09].

La base de données de structures de protéines la plus générale et de loin la plus fournie est la PDB (pour Protein Data Bank) [BKW+77]. Pour obtenir de telles structures de protéines, il faut dans un premier temps purifier la protéine à étudier, c'est-à-dire l'isoler de son environnement biologique. Cette étape est parfois complexe car elle nécessite des combinaisons de filtrages différents, chacun utilisant les propriétés physico-chimiques des composés biologiques présents pour réussir à les séparer.

Ensuite, il existe deux principales techniques obtenir la structure à l'échelle atomique : la diffraction de rayons X par un cristal de protéines ou la résonance magnétique nucléaire. La diffraction par rayons X est la méthode historique (1ère structure publiée en 1958 [KBD+58]). Elle consiste à créer un cristal avec de multiples instances de la protéine à résoudre, puis à observer la figure de diffraction générée par le passage de rayons X à travers le cristal. Un traitement mathématique permet, à partir de l'intensité des figures, de diffraction de calculer l'emplacement des nuages électroniques dans l'espace en trois-dimension. La principale limite de la méthode par cristallographie est le processus de création des cristaux. En effet, plusieurs facteurs peuvent empêcher la création des cristaux : flexibilité de la structure protéique, nécessité d'un environnement lipidique (pour les protéines trans-membranaires). C'est pourquoi, la RMN peut parfois résoudre des structures dont les cristaux sont difficiles à obtenir. Cette dernière fournit une indication des distances inter-atomiques au sein de la protéine. La RMN a aussi ses limites : plus la protéine est longue, plus le nombre de distances inter-atomiques similaires est grand et donc plus la détermination de la structure devient difficile. D'autre part, la RMN nécessite de pouvoir avoir des protéines solubles à de très fortes concentrations.

Ainsi, même si depuis plus de 50 ans, la technologie ne cesse de s'améliorer pour résoudre des structures tri-dimensionnelles de protéines de plus en plus précises, il s'agit toujours d'un processus long et délicat et certaines structures ne peuvent toujours pas être résolues.

1.2 Séquence-structure-fonction et évolution

Au fil des générations, les protéines *évoluent*. Une séquence protéique P d'un organisme va accumuler des mutations (des substitutions et des *indels* : des insertions et des délétions d'acides-aminés mais aussi de segments complets de séquence [TKF91]) et se diversifier en plusieurs séquences distinctes $P_a, P_b, ...$ On dit que ces séquences dérivées $P_a, P_b, ...$ sont des séquences *homologues* car elles dérivent du même ancêtre commun P. Comme nous l'avons vu précédemment, une séquence P_i (structure primaire) détermine

la structure tridimensionnelle de la protéine. A son tour, la structure tertiaire, c'est-à-dire le placement tridimensionnel des acides-aminés confère à la protéine une *fonction* spécifique dans l'organisme (la catalyse de réactions chimiques, régulation d'expression de gènes, maintien d'un compartiment biologique, etc.). La figure 1.5 résume schématiquement le lien de causalité entre la séquence, la structure et la fonction d'une protéine — également appelé paradigme séquence-structure-fonction.

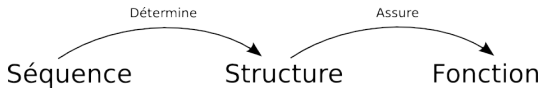

FIG. 1.5: Le paradigme séquence-structure-fonction

Ainsi, au fil des générations, les séquences et donc les fonctions des protéines *évoluent*. La pression de sélection exercée par l'environnement sélectionne les séquences codant pour des protéines dont la fonction procure un *avantage sélectif* à l'organisme. Ainsi, les séquences ou portions de séquence (comme un site catalytique par exemple) associées à des fonctions vitales seront conservées au cours de l'évolution voire seront améliorées (par exemple sélection des enzymes les plus efficaces pour les processus métaboliques centraux [BENS+11]).

On voit donc que la pression de sélection agit d'abord sur la fonction pour se traduire en conservation de structure et éventuellement de séquence. En effet, des séquences différentes pouvant coder pour une même structure, la séquence peut diverger raisonnablement sans que la structure ne soit affectée. Dans [CS96], il est par exemple estimé que si l'identité (voir section 2.1.0.1 pour une définition formelle) entre deux séquences est supérieur à 40%, les structures seront les mêmes, et que la structure commence à avoir des fortes modifications lorsque l'identité de séquence avoisine les 30% (zone appelée zone d'ombre ou *twilight zone* en anglais). Le fait que des séquences différentes aient une même structure peut s'expliquer soit parce que les mutations ont changé des acides-aminés par d'autres ayant les même propriétés physico-chimiques, soit parce que les interactions permettant le maintient de la structures sont redondantes et l'altération de quelques unes d'entre elles ne change pas la structure.

La pression de sélection peut même amener une fonction ou une structure à être totalement réinventée par une lignée d'organismes et ne partager aucune de similarité de séquence avec une protéine pré-existante de même structure : on parle alors d'*homoplasie*. Dans [Ros97], il est mesuré que des protéines partageant la même structure ont pour la plupart entre 8% et 9% d'identité de séquence, et qu'il est difficile de distinguer au sein d'un groupe structural si les séquences sont liées par homologie très lointaine ou par homoplasie.

Pour une même fonction, la séquence d'une protéine peut être très différente (homologie lointaine ou homoplasie) alors que sa structure sera davantage similaire. On résume généralement par le paradigme suivant :

la structure est plus conservée que la séquence.

Comme nous l'avons vu en introduction, connaître la fonction d'une protéine est un enjeu de haute importance pour la biologie moderne. On dispose donc de deux principales approches pour identifier une telle fonction : si on dispose de la structure, on cherche à reconnaître des structures spécifiques d'une fonction, et si on ne dispose que de sa séquence alors on peut soit vouloir retrouver une homologie (éventuellement lointaine) avec des protéines existantes dont on connaît la fonction, soit chercher à déterminer des caractéristiques structurales depuis la séquence qui seront autant d'indications sur sa potentielle fonction.

Une étape clé dans l'identification de la fonction d'une protéine est donc la capacité à identifier la similarité entre deux structures ou deux séquences. Ceci fait l'objet du chapitre suivant.

Chapitre 2

Mesures de similarité entre protéines

Comme nous l'avons vu dans la section précédente, les séquences des protéines à un temps t dérivent de séquences ancestrales à la suite de mécanismes de mutation. Ainsi, la probabilité que deux protéines homologues — *i.e.* dérivant d'un même ancêtre commun — aient une fonction identique dans l'organisme est d'autant plus grande que leurs séquences et qu'*a priori* leurs structures sont semblables.

Mesurer la similarité entre protéines tant au niveau de la séquence que de la structure permet de déterminer leur ressemblance fonctionnelle et ainsi de prédire la ou les fonctions d'une protéine à partir de protéines homologues (*i.e.* apparentées) de fonction connue.

Nous présenterons dans un premier temps l'alignement et la comparaison de séquences. Ensuite, nous présenterons les méthodes les plus courantes de comparaison de structures.

2.1 Comparaison de séquence

Comme nous l'avons vu dans la section 1.2, l'évolution des séquences protéiques est le produit de mutations sous la pression de la sélection naturelle. Ces mutations peuvent prendre la forme de substitutions d'acides-aminés mais aussi d'insertions ou de délétions de portions entières de séquences. Deux séquences seront considérées d'autant plus éloignées qu'elles auront subies d'évènements de mutation — potentiellement modificateurs de la structure et/ou de la fonction assurée dans l'organisme.

Ainsi, un moyen intuitif et biologiquement sensé de comparer deux séquences protéiques est de mesurer la quantité et l'importance de ces évènements d'édition. Techniquement, ceci est réalisé par l'alignement des séquences (quel acide-aminé d'une séquence A correspond à quel acide-aminé dans une séquence B) et l'évaluation d'un score rendant compte de leur proximité évolutive.

2.1.0.1 Alignement, identité, similarité

Formellement, un alignement entre d'une séquence $x_1...x_N$ sur une séquence $y_1...y_M$ est une fonction *partielle* injective $T : [\![1, N]\!] \to [\![1, M]\!]$. Ainsi $T(i) = j$ signifie que x_i est aligné à y_j. Un alignement est partiel car certains acides-aminés x_i ne seront alignés à aucun y_j, et injectif car deux acides-aminés x_i et $x_{i'}$ ne peuvent être alignés sur un même y_j. La plupart des alignements de séquence utilisés en bioinformatique sont des alignements croissants : on impose que $T(i) < T(j)$ pour tout $i < j \in dom(T)$. Un alignement croissant entre deux séquences est souvent représenté par la superposition de ces deux séquences en ajoutant un caractère (souvent "." ou "-") dit de "*gap*" lorsque celui-ci n'appartient pas au domaine de T (délétion) ou n'est pas dans l'image de T (insertion). Par exemple :

```
A  YMSDFAALTTFLRALGQYEIF.SDA.MDQLNSLITNYMDP
B  ..SDF..LTTFLRACGQYEIFDSDAMMDQLQSLLTNYMDP
```

FIG. 2.1: Les acides-aminés sur fond bleu représentent les acides-aminés identiques dans l'alignement : c'est-à-dire lorsque $x_i = y_{T(i)}$.

Le taux d'*identité* entre deux séquences $x_1...x_N$ et $y_1...y_M$ est le ratio $\frac{2}{N+M}\sum \mathbf{1}_{x_i=y_{T(i)}}$. Dans l'alignement de la figure 2.1, le taux d'identité est de $\frac{2}{38+36}34 \approx 91.9\%$.

Une première approche pour réaliser un alignement consiste à maximiser le taux d'identité de séquence. De manière à rendre compte plus finement de la distance évolutive entre les deux séquences à comparer, on peut aussi pénaliser le score par un coût attribué à chaque ouverture d'indel (*i.e.* insertion ou délétion d'un acide-aminé ou d'un segment d'acides-aminés) auquel on peut ajouter un coût proportionnel à la longueur de l'indel.

Optimiser un alignement selon cette dernière méthode ne permet pas de distinguer une substitution dont les propriétés physico-chimiques de l'acide-aminé est inchangée, d'une substitution transformant par exemple un acide-aminé chargé positivement en un acide-aminé chargé négativement. Une substitution préservant les propriétés physico-chimiques d'un acide-aminé aura plus de chances de ne pas modifier la fonction de la protéine. C'est pourquoi des scores de similarité entre acides-aminés ont été développés afin de générer des alignements plus fidèles à la distance fonctionnelle reliant les protéines à comparer.

Techniquement, un alignement T de deux séquences $x_1...x_N$ et $y_1...y_M$ sera obtenu en maximisant une fonction objectif définie comme le cumul des similarités des acides-aminés alignés duquel est déduit une pénalité pour chaque acide-aminé non aligné. Une telle fonction objectif F — dont le score optimal $F(T_{opt})$ est aussi appelé *similarité* de

séquence — peut être formalisé comme suit :

$$F(T) := \sum_{i \in dom(T)} s(x_i, y_{T(i)}) - \gamma \times (N - |\mathrm{dom}(T)|) \times (M - |\mathrm{img}(T)|) \tag{2.1}$$

où $s(x, y)$ est une mesure de similarité entre les acides-aminés x et y et γ le coût associé à un indel. On rencontre fréquemment une autre version de cette fonction objectif ayant des pénalités d'ouverture de *gap* (c'est-à-dire indépendant de la longueur de l'indel) de manière à rendre compte de l'insertion ou de la délétion de segments entiers de séquence.

Il existe différentes mesures de similarité s entre acides-aminés, les plus couramment utilisées étant PAM et BLOSUM, que nous détaillons dans les deux paragraphes suivant.

2.1.0.2 PAM

Les matrices PAM (pour *Point Accepted Mutation*) présentent la similarité entre acides-aminés comme la probabilité qu'un acide-aminé soit remplacé par un autre entre deux séquences partageant une très forte identité. Pour cela, 1572 séquences ont été groupées en 71 familles dont l'identité est au moins de 85% entre chaque séquence.

Les séquences au sein de chaque famille ont été regroupées sous forme d'arbre où chaque séquence est un nœud et dont les arrêtes relient les nœuds par un évènement de mutation. Ces arbres sont calculés de manière à minimiser le nombre total de mutations (principe de *parcimonie*).

De ces données sont déduites les nombres d'occurrences A_{ij} de la mutation d'un acide-aminé i en un acide-aminé j. Ces occurrences sont normalisées par des coefficients m_i de mutabilité relative (définis comme étant la fréquence de mutation le long d'une branche de l'arbre, divisé par le nombre d'occurrences de l'acide-aminé). On déduit alors une probabilité M_{ij} qu'un acide-aminé i soit muté en j après un évènement de mutation.

La matrice PAM considère d'autant plus similaires deux acides-aminés que ceux-ci ont été ponctuellement remplacé dans l'évolution des séquences. Cela permet effectivement d'attribuer des scores fiables lors de comparaison de séquences fortement apparentées, mais s'avère moins efficace lorsque les séquences possèdent une homologie plus faible [HH92]. En effet, au cours de l'évolution, lorsque l'homologie diminue entre deux séquences, les acides-aminés ne sont pas nécessairement remplacés un à un par un acide-aminé équivalent — comme cela est mesuré par une similarité PAM. Le paragraphe suivant présente les matrices BLOSUM qui permettent de définir une similarité entre acides-aminés de manière mieux rendre compte de l'homologie plus lointaine.

2.1.0.3 BLOSUM

Les matrices de substitution BLOSUMx [HH92] (où x représente le pourcentage d'identité de séquence utilisé dans la construction de la matrice) définissent la similarité entre acides-aminés de manière à mieux rendre compte de l'homologie lointaine entre deux séquences. Ces matrices sont construites plus simplement que les matrices PAM, à partir de blocs conservés au sein de l'alignement multiple (cf. section 3.1.1.1) de plusieurs

séquences homologues. La valeur B_{ij} correspond au log-ratio de la fréquence observée de substitution de i par j par la fréquence attendue si la préférence de mutation était uniquement dépendante de la fréquence d'apparition de l'acide-aminé j.

C'est généralement le type de similarité entre acide-aminé retenu pour les applications dont nous traiterons par la suite. L'outil Blastp (voir section 2.1.0.6) utilise par défaut BLOSUM62 comme matrice de substitution.

2.1.0.4 Needleman-Wunsch

[NW70] est un algorithme de programmation dynamique garantissant de trouver un alignement optimal pour une fonction objectif de la forme de celle de l'équation 2.1. Pour aligner deux séquences $x_1...x_N$ et $y_1...y_M$ par programmation dynamique, il faut de procéder en deux étapes. La première consiste à remplir une matrice $N+1 \times M+1$ dite de *scores* dont la cellule i,j contient le score maximal pouvant être obtenu en alignant l'acide-aminé i avec l'acide-aminé j. Ce score optimal est calculé récursivement par les équations suivantes :

$$\begin{array}{lcl} F_{0j} & := & \gamma \times j \\ F_{i0} & := & \gamma \times i \\ F_{ij} & := & \max(F_{i-1j-1} + s(x_i, y_j), F_{ij-1} + \gamma, F_{i-1j} + \gamma) \end{array} \tag{2.2}$$

où γ est un paramètre de l'algorithme représentant la pénalité attribuée à une insertion ou une délétion ponctuelle.

Dans un second temps, afin de déduire l'alignement optimal, l'algorithme part de la cellule de score maximal i_m, j_m, ce qui définit $T(i_m) := j_m$ puis on retrace le chemin ayant permis d'attribuer la valeur de chaque cellule : récursivement,

$$\begin{array}{lll} \text{si} & F_{i,j} = F_{i-1,j-1} + s(x_i, y_j) & \text{alors on attribue } T(i-1) := j-1, \\ \text{si} & F_{i,j} = F_{i,j-1} + \gamma & \text{alors on attribue } T(i) := j-1, \\ \text{si} & F_{i,j} = F_{i-1,j} + \gamma & \text{alors on attribue } T(i-1) := j. \end{array}$$

La similarité des séquences est mesuré par le score maximal de la matrice de score.

Lorsque plusieurs points de départ sont possibles (le score maximal de la matrice est présent dans plusieurs cellules), ou que plusieurs chemins sont possibles, cela donne lieu à des alignements différents ayant le même score.

Ces alignements sont dit *globaux* car ils optimisent l'alignement sur toute la longueur de la séquence.

2.1.0.5 Smith-Waterman

Comme nous l'avons vu dans la section 1.2, l'évolution exerce une pression de sélection sur les séquences protéiques. Il est fréquent que seuls quelques segments de séquence portant une grande pression de sélection soient conservés alors que le reste de

la séquence aura divergé au cours de l'évolution. Dans ce cas, un alignement global mettra en correspondance des parties de séquences faiblement apparentées sur une grande longueur plutôt que de mettre en correspondance seulement de petites parties très similaires. Une situation similaire peut se produire si deux protéines ne partagent qu'un seul des domaines les constituant : la similarité de séquence ne retrouvera que sur une petite portion de leurs séquences. Un algorithme comme celui de de Needleman-Wunsch génèrera alors des alignements peu représentatifs de la distance évolutive.

Dans [SW81], une variante *locale* de l'algorithme de Needleman-Wunsch — dit algorithme de Smith-Waterman — permet de mettre en correspondance de petits fragments de séquence très similaires, même si le reste de la séquence est très divergent. Techniquement, cet algorithme de programmation dynamique aligne au mieux les séquences en ne prenant pas en compte là où elles ne sont pas alignables. Formellement, cela se traduit en gardant uniquement les cellules positives de la matrice des scores, les autres étant mises à 0 :

$$\begin{aligned} F_{0,j} &:= 0 \\ F_{i,0} &:= 0 \\ F_{i,j} &:= \max\left[0, F_{i-1,j-1} + s(x_i, y_j), F_{i,j-1} + \gamma, F_{i-1,j} + \gamma\right] \end{aligned} \tag{2.3}$$

Bien que les deux algorithmes décrits ci-dessus soient relativement efficaces pour mesurer la similarité entre quelques séquences (leur complexité est en $\mathcal{O}(N \times M)$), ils ne sont cependant pas adaptés à la comparaison à l'échelle de bases de données entières (de l'ordre du million de séquence). On a alors recours à des heuristiques ne garantissant donc pas une solution optimale d'alignement mais qui se montrent beaucoup plus efficaces pour cette tâche. Blast, présenté dans le paragraphe suivant, est l'implémentation d'une heuristique rapide et précise qui en fait l'outil le plus utilisé pour les tâches d'alignement de séquence à grande échelle.

2.1.0.6 BLAST

BLAST (pour "*Basic local alignment search tool*" [AGM$^+$90]) est une heuristique de comparaison de séquence. Pour la recherche d'une séquence X dans une base de données B, l'algorithme est découpé en trois étapes :

- la constitution d'un ensemble de k-uplets (appelés *graines*, avec par défaut $k = 3$ pour les séquences protéiques) ayant un score de similarité supérieur à un certain seuil T avec X,
- la recherche *exacte* de ces graines dans la base de données B,
- extension de l'alignement autour des graines afin d'extraire le fragment maximisant le score de similarité local.

Blastp [CCA$^+$09] est l'implémentation de cet algorithme pour le traitement des séquences protéiques.

2.1.0.7 Significativité d'un alignement local

Dans [KA90], une méthode est décrite pour évaluer la significativité d'un score de similarité de séquence local. Cette significativité se traduit par une E-value, c'est-à-dire

par le nombre d'alignements locaux d'une séquence avec une base de données B ayant un score de similarité supérieur à un score S si les séquences en questions étaient des séquences aléatoires (*i.e.* ayant chaque acide-aminé tiré aléatoirement indépendamment à chaque position selon une distribution de fond).

Plutôt que de traiter directement la similarité de séquence par rapport au score d'alignement local, il est plus parlant d'utiliser la E-value. En effet, le nombre n d'alignements locaux ayant un score supérieur à S suit une distribution de poisson de paramètre E (la E-value correspondant à S), ce qui signifie que $P(n = n_0) = \frac{E^{n_0}}{n_0!}e^{-E}$. Ainsi, la probabilité p qu'il existe une séquence de B ayant un score $\geq S$ (*i.e.* que $n >= 1$) est donnée par $p = 1 - e^{-E}$. Aussi, pour des petites valeurs de E, on a au premier ordre $p \approx 1 - (1 - E) = E$.

Lorsque l'on cherche à savoir si deux séquences sont proches, fixer un seuil faible de E-value revient à fixer approximativement (au premier ordre) le même taux d'erreur de première espèce. Cependant, quand on recherche des séquences avec une homologie plus lointaine, il se peut qu'on ait à fixer des valeurs de E supérieures à 1.

2.2 Comparaison structurale

L'évolution des séquences protéiques se traduit par une évolution des structures. On peut ainsi identifier des similarités et classifier des protéines en les comparant uniquement d'un point de vue structural. De plus, il peut être nécessaire d'utiliser la comparaison structurale pour identifier la fonction de protéines. En effet, la comparaison de séquence n'est parfois pas suffisante pour attribuer une fonction à une protéine : d'une part il se peut qu'aucun homologue ne soit connu, et d'autre part il arrive que deux homologues proches n'aient pas la même fonction. La structure peut dans ces cas aider à identifier des protéines de structure similaire dont la fonction est connue [PO08].

Comme dans le cas des séquences, la plupart des méthodes établissent une correspondance explicite des acides-aminés (un alignement) permettant d'évaluer la similarité structurale sur les régions conservées et de pénaliser éventuellement les indels. Aussi, l'alignement est généralement défini en vue d'optimiser une certaine mesure de similarité structurale.

Nous exposerons ici quelques mesures de similarités (ou "scores") de structure et leurs éventuelles méthodes d'alignement associées. Nous utiliserons par la suite la comparaison structurale pour comparer des fragments de structures (*i.e.* portions de structure contigües en séquence). C'est pourquoi nous nous concentrons ici sur les méthodes qui peuvent être appliquées à la comparaison de fragments.

Les similarités de structures présentées dans cette section utilisent uniquement l'emplacement des atomes de C_α de chaque acide-aminé. Nous noterons $a_1, ...a_N \in \mathbb{R}^3$ les points représentant les atomes C_α d'une protéine A dans l'ordre de la structure primaire et $b_1, ...b_M$ d'une protéine B. On notera T un alignement entre A et B.

2.2.0.8 RMSD

Le moyen le plus simple pour évaluer la similarité de deux structures protéiques est le RMSD (pour *Root Mean Square Deviation*) : étant donné un alignement T des acides-aminés, il s'agit de calculer la déviation moyenne des distances entre les atomes C_α alignés suite à une superposition optimale (translation et rotation de corps rigide) des structures :

$$RMSD_{opt}(A,B) := \min_{F \text{isom. d.}} \left[\frac{1}{|dom(T)|} \sqrt{\sum_{i \in dom(T)} ||a_i - F\, b_{T(i)}||^2} \right] \tag{2.4}$$

où F varie sur l'ensemble des isométries directes, c'est-à-dire sur l'ensemble des compositions possible de translation et rotation de corps rigides[1].

Étant donné l'alignement T, il n'est pas nécessaire de calculer explicitement l'isométrie F, et on peut directement obtenir le RMSD optimal en $\mathcal{O}(|dom(T)|)$ en alignant les centres de gravités de A et de B puis en appliquant la formule [Kab76] :

$$RMSD_{opt}(A,B) := \sqrt{\frac{1}{|dom(T)|}(||A||^2 + ||B||^2 - 2(\sigma_1 + \sigma_2 + \sigma_3))} \tag{2.6}$$

où : A est la matrice $|dom(T)| \times 3$ dont la ligne i sont les coordonnées a_i pour $i \in dom(T)$, et similairement B est la matrice $|dom(T)| \times 3$ dont la ligne i sont les coordonnées $b_{T(i)}$ pour $i \in dom(T)$, et $\sigma_1, \sigma_2, \sigma_3$ sont les valeurs propres de la matrice 3×3 $A^t\, B$.

2.2.0.9 TM-Score

Une des critiques que l'on peut avoir par rapport au RMSD est sa sensibilité à la longueur des structures à comparer : plus le nombre d'acides-aminés est important, plus le RMSD a sa distribution décalée vers des valeurs plus élevées. De plus, s'agissant d'une déviation moyenne, un seul acide-aminé peut dégrader fortement le RMSD total — et même théoriquement le faire tendre vers l'infini— même si la superposition est parfaite sur tout le reste de l'alignement.

Le TM-Score (pour *Template Modeling* [ZS04]) permet de répondre à ces deux points : il est normalisé dans l'intervalle $[0,1]$ et attribue un score compris dans $]0, \frac{1}{M}]$ (où M est le nombre d'acides-aminés de la structure B) à chaque acide-aminé. Ainsi, un seul acide-aminé pourra "dégrader" le score total seulement à proportion de sa "participation" $\frac{1}{M}$ à la structure. Par ailleurs, le score de chaque acide-aminé donne plus de poids aux faibles déviations, comme on peut le voir sur la figure suivante :

[1] Par abus de notation, la valeur $RMSD_{opt}(A,B)$ est souvent notée $RMSD(A,B)$ alors qu'en toute rigueur $RMSD(A,B)$ représente la déviation entre les coordonnées des atomes C_α de chaque structure :

$$RMSD(A,B) := \frac{1}{|dom(T)|} \sqrt{\sum_{i \in dom(T)} ||a_i - b_{T(i)}||^2} \tag{2.5}$$

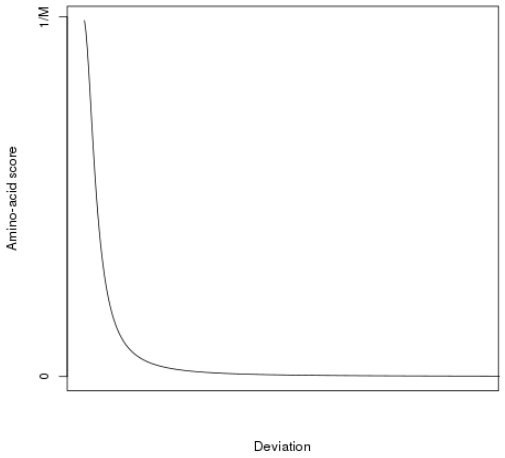

FIG. 2.2: Contribution d'un acide-aminé au TM-Score par rapport à sa déviation.

Formellement, le TM-Score est défini par l'équation suivante :

$$\text{TM-score}(A,B) := \max_{F \text{isom. d.}} \left[\frac{1}{M} \sum_{i \in dom(T)} \frac{1}{1 + \left(\frac{||a_i - F\, b_{T(i)}||}{Z(M)} \right)^2} \right] \tag{2.7}$$

où N_B désigne le nombre d'acides-aminés de B et $Z(M) := 1.24(M-15)^{1/3} - 1.8$ est un facteur empirique de normalisation. De même que pour le RMSD, le TM-score n'aurait pas réellement de sens si les structures n'étaient pas préalablement superposés de manière à maximiser le TM-score : c'est pourquoi "max" permet d'optimiser la valeur du TM-Score par superposition optimale des deux structures par une isométrie directe F.

Dans ce but, il existe un outil dédié appelé TM-align [ZS05] qui cherche un alignement T ainsi qu'une isométrie directe permettant de maximiser le TM-score. TM-Align est une heuristique rapide, mais qui n'assure pas un résultat optimal (voir des exemples concrets en section 5) comme pourrait le faire un algorithme exact (comme dans le cas du RMSD).

Après alignement avec TM-Align, un TM-score inférieur à 0.2 signifie que les deux structures ne sont pas apparentées, et un score supérieur à 0.5 signifie en général que les deux structures sont similaires (*e.g.* partagent le même *fold* pour le cas de protéines entières) [ZS05].

2.2.0.10 BC score

Plus récemment a été développé le score Binet-Cauchy (ou *BC score* [GT13]). Un des intérêts de ce score réside dans le fait qu'il permet de distinguer une structure et son miroir. En effet, le BC score peut être vu comme un produit scalaire normalisé entre deux vecteurs dont chaque coordonnées représente le volume (orienté) décrit par chaque les triplet de points de la structure. Ainsi, deux structures similaires décriront les mêmes volumes et le produit scalaire (normalisé) vaudra 1, et une structure et son miroir décriront des volumes orientés opposés et le produit scalaire vaudra -1. Ainsi, ce score permet de venir pallier à l'invariance par symétrie de certaines mesures de similarité de structure telles que celles présentées dans les prochains paragraphes.

Formellement, étant donné un alignement T entre deux protéines A et B, la définition formelle du BC score est la suivante :

$$BC(A,B) := \frac{\det(A^t B)}{\sqrt{\det(A^t A)\det(B^t B)}} \tag{2.8}$$

où : A est la matrice $|dom(T)|\times 3$ dont la ligne i sont les coordonnées a_i pour $i \in dom(T)$, et similairement B est la matrice $|dom(T)| \times 3$ dont la ligne i sont les coordonnées $b_{T(i)}$ pour $i \in dom(T)$. Du fait de la normalisation, les valeurs du BC score sont comprises dans l'intervalle $[-1, 1]$

Le BC score peut se calculer efficacement en temps linéaire par rapport $|dom(T)|$ [GT13]. De plus, il s'agit d'un noyau — *i.e.* il s'agit d'un produit scalaire entre vecteurs dans un espace de grande dimension — ce qui le rend particulièrement attractif pour des applications de classifications comme les Support Vector Machines [Gor04].

Enfin, les scores de similarité qui respectent l'inégalité triangulaire permettent d'avoir une meilleure qualité de classification [RF03, Koe01]. Or la valeur $d_{BC}(A,B) := 1 - BC(A,B)$ est une pseudo-distance respectant donc l'inégalité triangulaire, pouvant donc être utilisée comme mesure de similarité dans le cas de tâches de classification.

2.2.0.11 RMSD_d

Il existe un autre type de mesure de similarité de structure ne se focalisant pas uniquement sur des distorsions locales comme les scores précédents, mais prenant en compte l'ensemble des distances inter-atomiques des structures. La mesure la plus simple d'entre elles est le le RMSD_d (d signifiant distance) qui mesure la moyenne des déviations entre toutes les distances inter-atomiques des C_α. Sa définition formelle est la suivante :

$$\text{RMSD}_\text{d}(A,B) := \sqrt{\frac{1}{\binom{|dom(T)|}{2}} \sum_{i<j<n} (D_{Ai,j} - D_{Bi,j})^2} \tag{2.9}$$

où $D_{Ai,j} = ||a_i - a_j||$ et $D_{Bi,j} = ||b_{T(i)} - b_{T(j)}||$ pour $i, j \in dom(T)$. Les matrices D_A et D_B sont appelées *matrices de distances internes* ou simplement *matrices de distances*.

Cette mesure de similarité utilisant la déviation des distance internes, il n'y a pas lieu de chercher une superposition optimale des structures, les distances inter-atomiques des atomes de C_α étant bien entendues invariantes par isométrie. De ce fait, un des

désavantage à utiliser le RMSD_d , est qu'il est impossible de faire la distinction entre une structure et son miroir (le RMSD_d vaudra alors 0).

2.2.0.12 DALI

Le RMSD_d mesure la différence entre les distances internes des structures à comparer. Ainsi, une différence d'1Å sur une distance interne valant 5Å a autant d'importance pour le score global qu'une différence d'1Å sur une distance interne de 30Å. DALI [HP00] est une heuristique[2] d'alignement de matrices de distances internes qui se base sur un score pondérant différemment les différences de distance des paires alignées d'acides-aminés suivant la valeur de celle-ci. Formellement, le score associé à l'alignement d'une paire d'acides-aminés i, j d'une protéine A avec une paire k, l d'une protéine B vaut :

$$S_{i,j,k,l} := (0.2 - 2\frac{|D_{Ai,j} - D_{Bk,l}|}{D_{Ai,j} + D_{Bk,l}})e^{-|D_{Ai,j}+D_{Bk,l}|^2/1600} \tag{2.10}$$

Les paramètres 0.2 et 1600 sont fixés de manière *ad hoc* et on peut se poser la question de l'universalité de ceux-ci (suivant le type et la taille des protéines à comparer par exemple).

Le score global associé à un alignement est la somme des scores de chaque paire alignée $i, j \leftrightarrow k, l$.

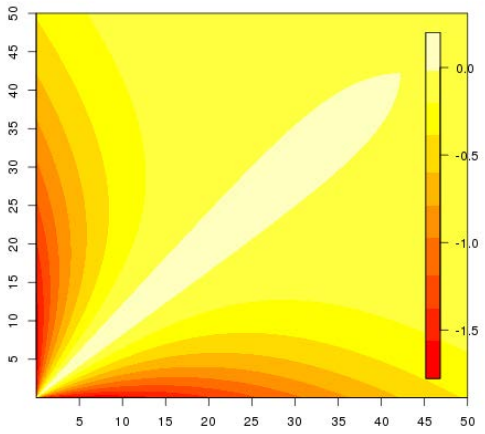

FIG. 2.3: Valeur de $S_{i,j,k,l}$ en fonction de $D_{Ai,j}$ et $D_{Bk,l}$.

2.2.0.13 Contact Map Overlap

On peut largement compresser l'information fournie dans la matrice des distances internes en une matrice binaire dite *matrice de contact* qui possède un 1 là où la distance

[2]Il existe maintenant un algorithme exact nommé DALIX [WAK12] permettant d'obtenir l'alignement optimal au prix d'un coup computationnel plus élevé.

interne est inférieure à un certain seuil σ et 0 sinon :

$$D^{\sigma}_{i,j} := \begin{cases} 1 \text{ si } D_{i,j} < \sigma \\ 0 \text{ sinon} \end{cases} \tag{2.11}$$

On dira que les acides-aminés i et j sont en contact lorsque $D^{\sigma}_{i,j} = 1$. En utilisant uniquement l'information fournie par cette matrice, on peut calculer l'alignement de structure maximisant le nombre de contacts communs dans l'alignement. Formellement, il s'agit de maximiser :

$$CMO(P,Q) := \sum_{i,j \in dom(T)} D^{\sigma}_{P\,i,j} \times D^{\sigma}_{Q\,T(i),T(j)} \tag{2.12}$$

Un algorithme exact rapide pour trouver un tel alignement est présenté dans [AMDY11].

L'information de contact dans une structure de protéine est caractéristique de la structure elle-même. On peut ainsi reconstruire une structure à partir de l'information de contact [DSS+10].

L'intérêt des contacts dans les structures de protéines va bien au-delà de la comparaison de structure. Par exemple, grâce à des développements récents, la prédiction de contact à partir de la séquence ([MCS+11, MCS+11, MHS12] et voir par exemple la section 3.1.2) semble s'établir comme un point clé intermédiaire vers la prédiction de structure de protéine à partir de la séquence [MDF+14].

Conclusion Nous avons vu comment comparer des protéines au niveau de leur séquence ainsi qu'au niveau de leur structure. Pour cela, la plupart des méthodes procèdent en deux étapes : une étape d'alignement établissant une correspondance entre les acides-aminés des deux protéines à comparer et une étape d'évaluation de similarité établissant un score rendant compte de la similarité issu de l'alignement. Cependant, l'alignement explicite des acides-aminés peut être source de problèmes. En effet, l'alignement est soit issu d'une superposition des structures en 3D (et donc ne permettra de comparer que des structures très similaires, car superposables), soit optimisé pour maximiser le score de similarité dont les effets de seuil liés à la décision d'aligner ou non un acide-aminé nécessitent d'introduire des paramètres *ad hoc* non universels. Enfin, les scores de similarités échouent à quantifier la (dis)similarité des parties non alignées, ce qui peut — suivant les applications — être fondamental pour caractériser la ressemblance de deux structures. En section 5, introduirons un nouveau score de similarité appelé ASD (pour *Amplitude Spectrum Distance* [GC15]) qui compare *globalement* des structures sans avoir recours à un alignement explicite des acides-aminés, qui respecte l'inégalité triangulaire et qui est tolérant aux indels.

Nous allons à présent voir comment représenter un ensemble de protéines similaires, et chercher à caractériser leurs similarités encore une fois au niveau de la séquence mais également de la structure.

Chapitre 3

Caractérisation d'un ensemble de protéines

Les protéines sont souvent regroupées sous forme de familles. Ces familles peuvent être de différente nature : fonctionnelle (partageant par exemple une même activité enzymatique), structurale (par exemple partageant un même repliement) ou encore séquentielles (par exemple des séquences homologues partageant un ancêtre commun). Lorsqu'un expert définit une famille il est intéressant de pouvoir la caractériser formellement : on cherche alors un trait commun de séquence ou de structure permettant d'identifier une protéine comme appartenant à la famille. La présence de cette signature dans une nouvelle séquence permet alors de prédire l'appartenance à la famille.

Dans la section 1.2 il a été mentionné que la pression de sélection sur les protéines n'est pas uniforme sur la structure ni sur la séquence : la conservation sur un ensemble de protéine est souvent localisée. Les portions de séquence conservées (type site catalytique) servent de base à la construction de *signatures* recensées dans différentes bases de données (Pfam [BBC+02], Prosite [SCC+13a], InterPro [MCD+14]). Par *motif*, nous désignerons largement toute description de séquence ou de structure ayant au moins deux occurrences au sein d'un ensemble de protéines et nous désignerons par *signature* un motif caractéristique d'une famille (*i.e.* partagé par tous les membres de la famille).

Au niveau de la structure, une signature peut par exemple prendre la forme d'un cœur commun et la variabilité des boucles entourant le cœur différencie chaque protéine au sein de la famille. Comme pour les séquences, des bases de données cherchent à définir et classifier des signatures caractéristiques de familles structurales [SLC+15, PSSC08].

Nous verrons dans un premier temps différentes méthodes pour caractériser une famille de séquences en se basant sur l'homologie et la conservation *locale* de segments de séquence. Nous verrons ensuite des méthodes qui permettent de détecter un signal structural *à travers la séquence*, et ainsi pouvoir caractériser une relation *structurale* entre les séquences de la famille. Enfin, nous aborderons des méthodes permettant de caractériser un ensemble de structures à travers la recherche de caractéristiques communes (*i.e.* de signatures structurales).

3.1 Caractérisation via la séquence

Le moyen le plus simple de caractériser une famille de séquences est d'identifier les conservations de séquence au sein de la famille. Pour cela on construit un alignement dit *alignement multiple de séquences* maximisant la similarité entre toutes les séquences de la famille.

3.1.1 La conservation

3.1.1.1 L'alignement multiple

Comme pour l'alignement de deux séquences (cf. section 2.1.0.1), on peut réaliser un alignement simultané de plusieurs séquences protéiques, que l'on désigne sous le nom d'*alignement multiple*. Nous présentons ici encore uniquement le cas classique d'alignements croissants, dont la représentation sous forme de matrice généralise le cas de l'alignement de deux séquences : les lignes représentent les séquences, les colonnes représentent les acides-aminés alignés. Les cellules de la matrice pouvant posséder un caractère spécial de *gap* "-" ou "." lorsqu'une séquence ne fournit pas d'acide-aminé à aligner. Par exemple :

```
A  ..TGA.VLANVPQS.KQRLKLEFANNNT.AFA.AV.GANPLEAIYQGAGAADCEF
B  ..TGA.VLANVPQS.KQRLKLEFANNNT.AFA.AV.GANPLEAIYQGAGAADCEF
C  PYSGASAIIDFRKYFNGDLDLTHAPSDSIEYDLALQNQDNVYSLYVSY.VLPY.Y


A  EEIS.YTV.YQSYLDQLPVG.
B  EEIS.YTV.YQSYLDQLPVG.
C  DQLAALPAQVAAIVQQYVARQ
```

Dans le cas d'alignement de deux séquences, l'objectif était de maximiser un score. La transposition de ce problème au cas de l'alignement multiple consiste à maximiser la somme des scores de toutes les paires de protéines (appelé communément le *SP*-score pour *Sum of Pairs Score*). On obtient une difficulté algorithmique de la classe NP-difficile [Eli03]. On peut cependant résoudre de façon approchée ce problème via des heuristiques (*i.e.* ne garantissant pas *in fine* un score global optimal) itératives optimisant progressivement l'alignement global. C'est le cas des outils les plus courants comme T-Coffee [NHH00], ClustalW [LBB+07], MAFFT [KMKM02] et MUSCLE [Edg04].

Un alignement multiple permet de repérer les conservations de séquence au sein d'une famille, point de départ aux méthodes présentées ci-après pour la générations de motifs caractéristiques.

3.1.1.2 Motifs PROSITE

Les motifs PROSITE [SCC+13a] permettent de caractériser de manière déterministe (à la différence des méthodes statistiques des prochains paragraphes) un enchaînement d'acides-aminés en autorisant des insertions et des caractères *jokers*. Par exemple le

motif C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H décrit toute sous-séquence commençant par un "C" suivi de deux, trois ou quatre caractères (*x(2,4)*), suivi d'un "C", puis de trois caractères, suivi d'un caractère parmi L,I,V,M,F,Y,W ou C, etc.

La base de données PROSITE [SCC+13a] référence des motifs caractéristiques de familles de séquences, structurales et fonctionnelles. Ces motifs sont généralement définis à partir d'un alignement multiple, mais il existe des outils de découverte de motifs tels que Pratt [Jon97] capables de générer un motif PROSITE (optimal pour une certaine fonction objectif) à partir d'une liste de séquences non alignées.

3.1.1.3 PSSM

Bien que les motifs PROSITE soient bien adaptés par exemple pour la reconnaissance de sites catalytiques nécessitant d'avoir exactement tel ou tel acide-aminé pour réaliser une liaison avec un ligand, il arrive que plus de souplesse soit tolérée dans les substitutions possibles et qu'il existe des choix préférentiels pour certains acides-aminés à certaines positions.

On peut décrire ces motifs avec une PSSM (pour *Position-Specific Scoring Matrix*). Une PSSM est une matrice définissant un motif de séquence en détaillant la distribution de ses acides-aminés position par position. Formellement, pour un motif de longueur N, il s'agit d'une matrice M de taille $20 \times N$ telle que l'élément m_{ij} est la probabilité d'occurrence de l'acide-aminé i à la position j dans la séquence. Cette probabilité peut-être directement déduite en regardant la distribution empirique d'un alignement multiple, ou bien complétée à l'aide de pseudo-comptages (technique consistant à ajuster les probabilités de chaque entrée de manière à éviter d'avoir des entrées nulles).

Généralement, on rapporte la probabilité d'occurrence à une distribution de fond en acides-aminés qui peut être choisie uniforme ou *ad-hoc* selon le contexte. Aussi, pour des raisons usuelles d'additivité du score, on utilise le *log* de cette probabilité normalisée : $m_{i,j} := \log(p_{ij}/b_i)$ où b_i est la distribution de fond. On peut ainsi déduire le score d'une séquence $A := a_1...a_N$ par rapport à un motif M avec la formule : $s(A) := \sum_{j=1}^{N} m_{a_j,j}$.

3.1.1.4 HMM

Les PSSM ne permettent pas de modéliser les indels dans les séquence, comme on peut le faire avec les motifs PROSITE.

Les profils HMM (pour *Hidden Markov Model*) combinent les avantages des PSSM et des motifs PROSITE : ils permettent de caractériser des motifs en autorisant des indels tout en émettant un score rendant compte de l'adéquation d'une séquence avec le motif défini par le HMM.

Un profil HMM est un HMM avec une structure prédéfinie possèdant des état dits de *match* correspondent aux colonnes d'une PSSM, mais aussi des états d'insertion et de délétion dont les probabilités de transition entre chaque état sont inférées à partir d'un alignement multiple par maximum de vraisemblance avec des outils comme HMMER [Edd98].

3.1.2 Corrélations séquentiellement distantes

La conservation locale n'est pas nécessairement la seule trace au niveau de la séquence de l'appartenance à une famille structurale. En effet, d'une part, comme nous l'avons déjà vu, la structure est souvent plus conservée que la séquence, et il se peut que deux séquences aient fortement divergé au point de n'avoir que 30% d'identité de séquence, mais ayant toujours la même structure [CS96]. Par ailleurs, comme nous l'avons vu dans la section 1.2, la pression de sélection peut faire évoluer deux séquences non-homologues (ne partageant pas un ancêtre commun) vers une même structure et a fortiori une même fonction.

La similarité de séquence peut donc ne pas rendre compte de parties plus conservées que d'autres — et la conservation globale pourrait même être très faible —, mais les structures associées peuvent être très similaires.

Un moyen de caractériser des familles structurales sans avoir recours directement à l'homologie de séquence est d'observer le fait suivant : dans la protéine, il existe des acides-aminés nécessaires au maintien global de la structure via l'interaction physico-chimique de certains acides-aminés. Ainsi, les paires d'acides-aminés proches en structure portent une contrainte mutuelle en séquence. Par exemple, si un acide-aminé est muté alors qu'il participe à une interaction essentielle au maintien de la protéine, il y a de fortes chances qu'une mutation dite *compensatoire* soit sélectionnée pour l'acide-aminé avec lequel il interagissait de manière à préserver l'interaction entre les deux.

On voit alors que la famille structurale à laquelle appartient une séquence est caractérisé notamment par des interactions séquentiellement distantes. On se propose ici de présenter des méthodes permettant la caractérisation d'une famille structurale à l'aide de l'information de co-évolution en séquence, puis nous introduirons des techniques plus récentes permettant de prédiction de contact à partir de cette même information. Enfin, nous verrons comment on peut évaluer l'adéquation d'une séquence à une structure.

Ensembles de co-évolution Dans [HRLR09], les auteurs présentent une méthode utilisant un alignement multiple de séquences protéiques d'une même famille pour déterminer des ensembles d'acides-aminés — appelés Protein Sectors — ayant une histoire évolutive commune. L'idée fondamentale derrière cette approche est de regrouper les positions de co-évolution de résidus.

Techniquement, les auteurs calculent la matrice d'entropie relative entre chaque colonne de l'alignement multiple des séquences de la famille. Ils calculent ensuite les vecteurs propres de cette matrice et conservent ceux associées aux valeurs propres les plus significatives[1]. L'espace des vecteurs propres est ensuite découpé par des frontières linéaires en "secteurs" (la technique est laissée libre dans la méthode présentée, bien que les auteurs aient choisi d'utiliser l'Independent Component Analysis [HKO04]), qui détermine l'appartenance de chaque résidu à un unique secteur. Un secteur va donc regrouper des acides-aminés ayant une histoire évolutive commune.

[1] A noter que le mode dominant (correspondant à la valeur propre la plus haute) est mis de côté car représentant uniquement selon les auteurs la parenté phylogénétique globale de la famille et ne rendant donc pas compte d'une unité ayant une histoire évolutive propre au sein de la famille.

On observe que les secteurs calculés dans les exemples présentés dans l'article [HRLR09] présentent une proximité spatiale surprenante avec une identification fonctionnelle relativement claire (comme par exemple un secteur représentant la poche du site catalytique, un autre contenant la triade catalytique elle-même), bien qu'aucune contrainte de ce type n'ait été intégré dans le calcul des secteurs. Ceci indique que la co-évolution est liée à la structure tri-dimensionnelle de la protéine : afin de maintenir la fonction — et donc la structure — la protéine, des mutations compensatoires sont sélectionnées au cours de l'évolution.

L'approche de *Protein Sectors* nécessite cependant un alignement multiple réalisé avec beaucoup de séquences afin d'avoir une statistique fiable pour l'entropie relative des colonnes de l'alignement. *"Blocks In Sequences" (BIS)* [DC12] contourne cette difficulté en utilisant une méthode combinatoire qui identifie en premier lieu les fragments (contigus en séquence) qui sont conservés dans l'alignement multiple puis qui regroupe ces fragments afin de détecter ceux qui co-évoluent. Ici encore, bien qu'aucun *a priori* sur la proximité spatiale des résidus n'est été utilisé dans la définition des BIS, on observe que ceux-ci sont fortement localisés spatialement. Il est à noter que dans ces deux méthodes, les signatures définies ne sont pas contiguës dans la séquence.

Si on dispose d'une quantité suffisante de séquences d'une famille alors les ensembles d'acides-aminés définis par les deux méthodes précédentes donnent une information sur la proximité spatiale de ceux-ci et peuvent constituer une signature séquentielle du *fold* de la famille. Dans le prochain paragraphe, nous verrons comment on peut en effet aller plus loin et prédire de manière plus précise les résidus qui seront proches en structures.

Prédiction de contacts Comme nous l'avons vu avec les deux méthodes précédentes, les ensemble d'acides-aminés co-évoluant sont proches en structure. Pour prédire les contacts de manière précise (et non pas seulement des groupes d'acides-aminés proches en structure), on se base sur les même techniques que précédemment mais en exploitant une idée supplémentaire. Considérons trois résidus éloignés dans la séquence se retrouvant proches en structure dans une configuration comme dans la figure ci-dessous :

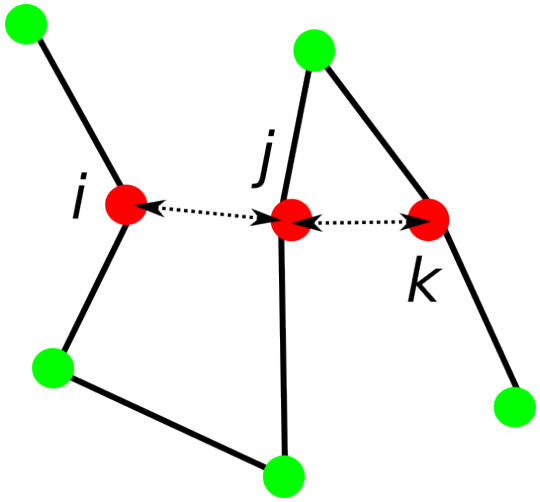

FIG. 3.1: Chaîne de contact induisant des co-variations entre les acides-aminés aux positions i et j, ainsi que j et k sur les séquences de la même famille structurale.

Alors au sein de cette famille structurale (on entend ici les protéines partageant le même *fold*), les séquences porteront des co-variations entre les résidus i,j et k. Cependant, en utilisant simplement l'information de co-évolution comme précédemment, on ne peut distinguer si i est en contact direct avec j ou avec k. C'est pourquoi certaines méthodes modifient la matrice de co-évolution afin d'obtenir une nouvelle matrice rendant compte plus finement des co-évolutions directes (*i.e.* une matrice décrivant les co-évolutions avec le plus de parcimonie possible : la donnée de co-évolution de i et j ainsi que j et k indique transitivement la donnée de co-évolution de i et k). Par exemple, PsiCov [JBCP12] est un outil de prédiction de contact qui utilise la matrice de corrélation partielle des occurrences des acides-aminés dans les colonnes d'un alignement multiple.

D'autres techniques similaires comme celle utilisée dans le prédicteur de *fold* EVfold [MCS+11] génèrent un modèle probabiliste d'entropie maximum pour les séquences d'un alignement multiple en respectant les distributions d'acides-aminés colonne par colonne ainsi que les distributions sur les paires de colonnes des paires.

Threading Nous venons de voir que la séquence peut rendre compte d'informations sur la structure : d'une part les fragments de séquence conservés représentent des fragments de structures conservés, mais aussi que la co-variation de résidus dans la séquence donne une information supplémentaire sur la structure que ces séquences décrivent.

On présente ici le *threading* [PMBB00] (littéralement *enfilage*), une méthode permettant l'identification du fold[2] associé à une séquence. Il s'agit d'aligner une séquence S sur une structure P (en général il s'agit du cœur structural partagé par toutes les

[2]Nous dirons ici que deux protéines partagent le même *fold* si elles se superposent sur une partie significative — notamment en dehors des boucles — de leur structure.

protéines d'un même *fold*) et d'en calculer un score d'adéquation. Un calcul de significativité permet de déterminer si la séquence S code pour une protéine ayant le *fold* P. On peut donc voir le threading comme une méthode de recherche d'homologues lointains dont la séquence ne porte pas assez de similarité pour identifier une homologie, mais dont les mutations préservent les interactions entre les acides-aminés.

Par exemple, dans l'outil FROST [MPZG02], pour chaque type de fold, un représentant de structure est choisi à partir duquel on extrait un cœur structural. Celui-ci consiste simplement à ne conserver que les acides-aminés inclus dans les structures secondaires hélices α et brins β.

FROST calcule ensuite deux scores (un score dit 1D d'homologie, et un score dit 3D de compatibilité d'acides-aminés en contact) dont l'optimisation guide l'alignement de la séquence sur le cœur.

Pour le score 1D, à partir de séquences homologues trouvées avec l'outil PSI-BLAST [AMS+97], FROST construit un alignement multiple et génère une PSSM pour chacun des fragments de structure secondaire constitutifs du cœur structural. Ceci permet d'établir un score de similarité de séquence comme nous l'avons vu précédemment avec dans la section PSSM.

Par ailleurs, FROST établit un score 3D d'interaction[3] : le score attribué à l'alignement de deux acides-aminés a et b aux positions i et j en contact dans le cœur est log-proportionnel à la probabilité jointe de trouver à des positions en contact le couple a, b ainsi que le couple r_i, r_j (où r_i, r_j désigne les acides-aminés du cœur en position i, j respectivement).

Ainsi FROST rend compte à la fois de la similarité de séquence liée à une éventuelle homologie, mais aussi à l'information de maintien de structure via le score 3D favorisant les acides-aminés interagissant en général.

Le *threading* permet de caractériser depuis la séquence l'appartenance à un *fold* en utilisant des informations de structure. Nous allons voir maintenant comment caractériser uniquement à l'aide de la structure l'appartenance d'une protéine à un fold ou plus généralement à une famille structurale.

3.2 Caractérisation structurale

Nous avons vu que la structure est généralement mieux conservée que la séquence. Ainsi, deux structures similaires peuvent avoir des séquences totalement différentes sans aucune conservation. Il devient alors difficile d'identifier une signature de séquence d'une famille structurale. Même des méthodes plus avancées telles que celles utilisant la coévolution d'acides-aminés ne peuvent pas toujours s'appliquer car elles nécessitent une quantité importante de séquence.

C'est pourquoi il est intéressant de chercher à caractériser une famille structurale de protéine uniquement à l'aide de l'information de structure. On peut alors chercher à

[3]Les deux types de score — score de PSSM et score d'interaction — tiennent compte du contexte (hélice, brin, coude) des résidus en question.

identifier des portions de structure conservées pouvant établir une signature structurale de la famille. Cette signature structurale peut servir de base pour définir des signatures séquentielles (par exemple définir des cœur pour le threading) ou servir directement à identifier la fonction d'une protéine.

3.2.1 Signature caractéristique d'une famille

Au même titre que certains segments de séquence sont fortement conservés au sein d'une famille, la conservation de fragments de structure peut indiquer une appartenance à une famille fonctionnelle : par exemple un site catalytique chez les enzymes.

Il s'agit donc de déterminer une sous-structure qui est significativement sur-représentée dans une famille de protéines.

Une approche de type "global" appelée FoldMiner [SB04] consiste à aligner les structures avec un outil dédié LOCK2 [SB04] et ne conserver que les alignements dont le score est significativement élevé. Le cœur structural (ou *core fold* en anglais) constitué de l'ensemble des fragments de structure partagés par une famille peut-être considéré comme une signature structurale de la famille : si cette structure est retrouvée dans une autre protéine, alors celle-ci sera considérée comme appartenant à la même famille structurale.

LOCK 2 [SB04] aligne en premier lieu les structures secondaires. Lorsque les structures sont très divergentes, LOCK 2 est capable de trouver des similarités locales, mais selon les auteurs, les alignements réalisés sont bien souvent globaux et les motifs locaux échappent à FoldMiner.

Afin de caractériser des similarités locales de structure, même pour des protéines ayant largement divergé et pouvant difficilement se superposer, une méthode consiste à avoir une représentation de la structure à l'échelle locale telle que présenté dans le prochain paragraphe.

3.2.2 Caractérisation locale

La diversité structurale à l'échelle très locale (quelques résidus) est limitée par des contraintes physiques d'interaction électro-magnétique. Les conformations locales du squelette protéique sont donc relativement peu nombreuses et ouvrent la voie à la construction d'une librairie des fragments de taille fixe telle que toute conformation du squelette peut être approximée par un élément de cette librairie. La structure peut ensuite être décrite par la séquence des fragments de la librairie.

La méthode la plus directe consiste à utiliser toutes les conformations possibles des fragments de longueur fixe comme c'est le cas de Rosetta [SKHB97b]. Rosetta a pour but de prédire la structure *de novo* à partir de la séquence et utilise un modèle probabiliste pour déterminer quels sont les fragments de structures les plus vraisemblablement associés à un fragment de séquence. Les auteurs décrivent ensuite une méthode d'assemblage de ces fragments structuraux via un mécanisme de recuit simulé.

Cette librairie peut en réalité être construite de manière plus concise afin d'utiliser une représentation symbolique des fragments (*i.e.* attribuer un symbole à chaque élément de la librairie). C'est le cas par exemple de [LBG+08] qui définie une librairie de

manière à avoir une redondance minimale selon des critères provenant de la théorie de l'information. Les auteurs arrivent à obtenir de meilleures prédiction de structure que ROSETTA avec une librairie bien plus concise. Dans le cas d'une telle librairie de petite taille, on parle généralement d'*alphabet structural* [SKHB97a, CGT04, EBHdB05, LBG$^+$08, dBEH00a], où les symboles composant l'alphabet sont des fragments locaux de structures.

Quelques exemples d'alphabets structuraux Dans [XCLM13] les auteurs cherchent à obtenir un maximum d'efficacité lors de l'assignation d'une lettre de l'alphabet à un fragment de structure. En effet, ils définissent une lettre de l'alphabet structural comme la discrétisation des distances internes d'un fragment de longueur fixe. Ainsi, en calculant les distances inter-atomiques des atomes C_α d'un fragment, on obtient directement le code de la lettre structurale. Ceci permet de décrire très rapidement les lettres structurales composant une structure et de pouvoir effectuer des comparaison structurales à grande échelle en comparant simplement le code des lettres les composant. Cette approche est intéressante d'un point de vue applicatif mais ne permet pas d'avoir une description à la fois concise et explicative d'une structure de protéine. En effet, soit la discrétisation est fine et le nombre de lettres de l'alphabet est très élevé pour une bonne qualité de description, soit on grossit le pas de discrétisation et la qualité de la description de la structure devient grossière.

Dans [DBEH00b], les auteurs décrivent un alphabet structural basé sur l'apprentissage de SOM (pour *Self-Organizing Maps*) représentant l'enchaînement des angles internes dans des fragments de taille fixe de 5 acides-aminés. Après une phase de convergence des SOM (chaque carte représentant une lettre de l'alphabet, appelées PB pour *Protein Blocks*), l'enchaînement des PB est modélisé par une matrice de transition. Le jeu d'apprentissage est alors décrit en terme de PB, en sélectionnant à chaque position le PB ayant la plus forte probabilité d'apparaître connaissant le PB précédent. Enfin, une dernière étape pour éliminer l'éventuelle redondance entre PB consiste à agglomérer les PB ayant des données de transitions ainsi que des descriptions d'angles diédraux proches.

La définition et l'enchaînement des lettres structurales peuvent être définis simultanément dans une même étape d'apprentissage [CGT04]. Cet alphabet structural est défini en modélisant avec un HMM l'enchaînement des fragments de longueur fixe de 4 résidus. Chaque état caché du HMM émet un vecteur à 4 dimension comprenant la longueur des 3 vecteurs inter-atomiques — entre chaque paire de C_α non-consécutifs *i.e.* 1-3,2-4,1-4 — ainsi que le déterminant décrit par ces 3 vecteurs, le tout selon une loi gaussienne multivariée à 4 dimensions dont les paramètres sont propres à chaque état caché. L'inférence des HMMs conduit à l'apprentissage de ces paramètres ainsi qu'à l'apprentissage des probabilités de transition. Un optimal est trouvé pour un HMM à 27 états selon le BIC (pour Bayesian Information Criterion) — mesure rendant compte de l'adéquation d'un modèle à des données relativement à la parcimonie de la description. Cette approche permet d'obtenir une compatibilité entre les différents états capturant les transitions entre chacun d'entre eux.

Ensemble de fragments Pour chaque fragment de longueur fixe d'une structure, on peut associer la lettre structurale de l'alphabet la plus similaire et ainsi on peut représenter la structure entière à travers l'ensemble des lettres structurales la composant. On obtient un vecteur dont les coordonnées sont le nombre d'occurrences de chaque symbole (le nombre de dimension du vecteur est donc la taille de l'alphabet structural). On peut alors comparer les structures ou classer les structures en famille en classifiant ces vecteurs [LPK09, BTNK10].

Séquence de fragments En utilisant uniquement un ensemble de fragments pour la comparaison structurale comme exposé dans le paragraphe précédent, on perd toute l'information relative à l'enchaînement des fragments est passée outre. On peut donc raffiner cette technique en représentant une structure par sa séquence de symboles structuraux, ce qui permet d'utiliser les techniques de comparaison de séquence afin de comparer des structures. Par exemple on peut définir des matrices de substitution de symboles structuraux comme dans [TGS+06] et ainsi détecter les fragments de structure conservés en réalisant des alignements locaux. Si on veut être plus expressif au niveau de la description d'une conservation structurale on peut par exemple utiliser un HMM pour modéliser la séquence de symboles structuraux et ainsi de caractériser une famille structurale [LPK09].

La similarité locale détectée avec les alphabets structuraux permet seulement de caractériser une similarité de structure contiguë dans la séquence. En effet, si la caractéristique d'une famille structurale réside dans la *position relative* de certains acides-aminés qui sont éloignés dans la séquence, alors il sera impossible de la capturer avec la description d'une structure via sa séquence de symbole structuraux. Pour caractériser de tels motifs, on a recours à des motifs structuraux non-contigus en séquence.

3.2.3 Motifs structuraux non-contigus

Étant donné la diversité possible des structures en jeu si on considère toutes les structures possibles localisées spatialement — *i.e.* non nécessairement contiguës en séquence —, il semble difficile de déterminer exhaustivement les motifs sur-représentés ; définir le "type" de motif structural recherché afin de limiter l'espace de recherche semble plus adapté. Par exemple, dans [JECT02], les auteurs définissent un motif comme étant un ensemble d'acides-aminés individuels proches en structure et dont la conformation est retrouvée dans au moins deux protéines d'une famille. Les sous-structures correspondant à ces motifs sont donc localisées dans l'espace mais potentiellement dispersées dans la séquence. Ceci permet de caractériser structuralement une portion locale conservée de structure, mais comme les acides-aminés sont dispersés dans la séquence, la significativité de leur identification en séquence sera toujours relativement faible et il est alors difficile de pouvoir identifier ces motifs depuis la séquence.

Conclusion L'identification de structures caractéristique peut être réalisé à un niveau global par alignement structural de protéines d'une même famille. Lorsque la divergence

des structures est trop importante, les caractéristiques locales sont difficiles à identifier par une telle approche globale. Dans ce cas, il est intéressant de caractériser la famille à l'échelle du fragment, par exemple en utilisant les alphabets structuraux. Ces derniers peuvent caractériser des similarité locales récurrentes, ou identifier des fragments de structure conservés, mais ils ne peuvent pas capturer les portions de structures caractéristiques dont l'éloignement en séquence est élevée. On peut alors définir des motifs spatialement locaux pour capturer de telles conformations éloignées le long de la séquence. La trace séquentielle de ces motifs se retrouve dispersée rendant difficile leur identification depuis la séquence.

C'est pourquoi avons introduit les fragments en contact (ou CF pour *contact fragments* en anglais, voir section 6), qui concilient la localité spatiale d'un contact structural avec son voisinage en séquence. Nous verrons qu'ils permettent d'identifier des structures caractéristiques dont l'éloignement en séquence peut être important, et que de simples modèles basés sur l'homologie permettent de les détecter depuis la séquence.

Deuxième partie

Contributions

Chapitre 4

Vers une formalisation du lien séquence-structure

Ce court chapitre propose un cadre reliant localement séquences et structures dans les protéines qui souvent guidé les intuitions, définitions et expérimentations présentés dans cette thèse. Nous introduisons notamment la notion de fragment *prédictible* qui a constitué les prémisses de l'introduction des fragments en contact (voir chapitre 6).

4.1 Cadre général pour le lien séquence-structure

Depuis l'expérience d'Anfinsen, on admet généralement que la séquence d'une protéine définit sa structure tri-dimensionnelle. Beaucoup de méthodes cherchent à utiliser ce principe pour prédire globalement la structure d'une protéine en prédisant celle-ci à une échelle locale, puis en assemblant les morceaux prédits (en utilisant par exemple des alphabets structuraux, présentés dans la section 3.2.2).

On peut se demander quelle est la plus petite échelle à laquelle une séquence possède une structure unique. Est-ce qu'il existe des motifs séquentiels (non nécessairement contigus) qui permettent de réduire cette échelle ? Cette échelle est-elle universelle pour toutes les séquences ? Pour cela nous introduisons un cadre formel permettant la définition de la notion de fragment *prédictible*.

On sait déjà qu'à une échelle (trop) locale, un même segment de séquence — *i.e.* une sous-chaîne contiguë — peut se retrouver dans différentes structures en adoptant des conformations différentes (figure 4.1). Réciproquement, un fragment de structure peut être codé par différentes séquences (figure 4.2).

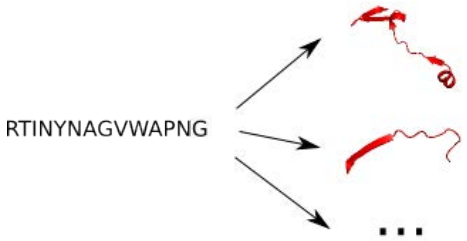

FIG. 4.1: Un segment de séquence peut coder différentes structures.

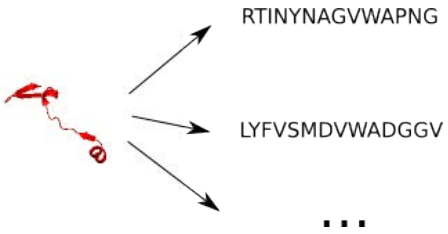

FIG. 4.2: Un fragment de structure peut être codé par différentes séquences.

On peut représenter ce lien séquence-structure par une relation binaire : on notera $q \sim t$ s'il existe une protéine possédant la sous-s**t**ructure t codé par la sous-sé**q**uence q.

Nous supposerons que nous nous plaçons dans un univers Ω_t de fragments de structures connues (comme par exemple l'ensemble des fragments des protéines de la PDB), et nous noterons $\Omega_q := \{q \in \Sigma^* | \exists t \in \Omega_t \text{ tel que } q \sim t\}$ l'univers des segments de séquences de structure connue où Σ est l'alphabet constitué par les 20 symboles des acides-aminés.

L'ensemble des structures associé à un ensemble de séquences $Q \subseteq \Omega_q$ sera noté $Q' := \{t \in \Omega_t | \exists q \in Q \text{ tel que } q \sim t\}$, et symétriquement, l'ensemble des séquences codant pour un ensemble de structures $T \subseteq \Omega_t$ sera noté $T' := \{q \in \Omega_q | \exists t \in T \text{ tel que } q \sim t\}$. On peut définir un opérateur de préfermeture $\widehat{T} := T''$. [1]. Un fragment de structure qui serait uniquement associé à un ensemble de segments de séquence se formalise alors ainsi $\widehat{\{t\}} = \{t\}$. Afin de prendre en considération le fait que les protéines sont relativement flexibles, et que deux structures peuvent être considérées comme identiques même si elles ont de petites variations, on considère que l'espace des structures est muni d'une mesure de dissimilarité d, et on définit cette relation d'unicité "à δ près" :

[1] En définissant récursivement la suite $T_0 := T$ et $T_{k+1} := T_k''$, on peut même définir un opérateur de fermeture [Rig48] sur l'ensemble des structures par $\overline{T} := \bigcup_{k \in \mathbb{N}} T_k$, induisant une topologie sur l'ensemble des structures. L'ensemble des fermés est alors $\{\overline{T} | T \in \Omega_t\}$.

Définition 4.1. *Pour une dissimilarité de structure d et un seuil δ, un fragment de structure t sera dit* fortement prédictible *si* $\sup_{x \in \widehat{\{t\}}} d(t, x) \leq \delta$.

La figure 4.3 représente schématiquement la relation entre la séquence et la structure d'un fragment fortement prédictible.

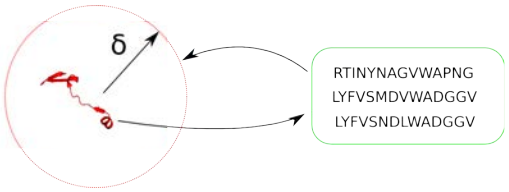

FIG. 4.3: les structures $\widehat{\{t\}}$ codées par les séquences $q \in \{t\}'$ codant pour le fragment t sont à une distance au plus δ de t. En identifiant les structures à δ près, on dira qu'un fragment prédictible est un fragment tel qu'aucune séquence codant pour ce fragment ne code pour un fragment différent.

Les fragments fortement prédictibles permettent une identification complète et fiable de structure locale depuis la séquence. En effet, si un fragment fortement prédictible t est structuralement caractéristique d'une famille F, alors on obtient immédiatement une signature séquentielle : toutes les protéines possédant t en structure locale sont dans F et leur séquence possède donc une sous-chaîne de $\{t\}'$, et réciproquement toute séquence de protéine possédant une sous-chaîne contenue dans $\{t\}'$ aura une structure locale similaire à t (à δ près) et la protéine appartiendra alors F.

Cependant, cette caractérisation ne considère que l'espace des fragments de structure connue et on peut vouloir généraliser la détection d'un fragment fortement prédictible dans des séquences de structure encore inconnues. Un cadre formel décrivant des propriétés théoriques de modèles de séquences permettant de généraliser la détection d'une structure est donné dans la prochaine section.

4.2 Cadre pour les modèles de détection de structure depuis la séquence

Si une séquence possède une sous-chaîne appartenant à l'ensemble t', alors d'après la définition d'un fragment fortement prédictible, la structure associée sera t (à δ près). Un modèle de séquence (par exemple HMM, motif PROSITE, etc.) associé à l'ensemble t' permettra d'une part une description plus explicative (identification des éventuels acides-aminés conservés par exemple) des séquences associées à t, mais pourra éventuellement généraliser l'identification de la structure t à des séquences de structure inconnue (*i.e.* n'étant pas dans Ω_q). Il est intéressant de distinguer deux propriétés différentes que ces modèles peuvent respecter : la *cohérence* et la *complétude*.

Pour tout fragment de structure $t \in \Omega_t$, on veut avoir un modèle M_t capable de reconnaître les séquences $\{t\}'$. On notera $M_t \vdash q$ dès lors que le modèle M_t reconnaît la

séquence q. Afin que le modèle soit spécifique du fragment qu'il décrit, on impose deux conditions :

Cohérence $M_t \vdash q \Rightarrow (\forall u \in \Omega_t, q \sim u \Rightarrow d(u,t) \leq \delta)$

Complétude $q \sim t \Rightarrow M_t \vdash q$

La cohérence assure que le modèle n'est pas trop général, *i.e.* qu'il ne décrit pas des séquences qui codent pour une structure différente. La complétude assure que toutes les séquences connues pour coder une structure sont effectivement décrites par le modèle. Par la suite, on dira simplement que t est *prédictible* par M_t si celui-ci est seulement cohérent.

4.3 Perspective : prédiction de structure par des structures locales

Prédictibilité partielle Une protéine entière ou même un long fragment protéique peut posséder des régions qu'il ne sera jamais possible de capturer par la prédictibilité (par exemple des régions désordonnées). On peut ainsi vouloir généraliser la prédictibilité de fragments à des motifs plus complexes. Par exemple en segmentant une structure en un ensemble de fragments, on peut redéfinir Ω_t comme l'univers des fragments extraits d'une même structure de protéine et définir pour un ensemble Q de séquences l'ensemble de structures $Q' := \{(t_1, ..., t_n) \in \Omega_t | \exists (q_1, ..., q_n) \in Q$ tels que $q_1 \sim t_1, ...$ et $q_n \sim t_n\}$. Les autres définitions telles que la cohérence et la complétude se redéfinissent de la même manière. On peut ainsi obtenir des structures partiellement prédictibles même pour des protéines possédant des régions désordonnées.

Couverture Lorsqu'on s'intéresse à prédire la structure d'un long fragment (ou même d'une protéine entière) à partir d'éléments locaux (de fragments par exemple), le Graal serait de disposer d'une libraire de fragments prédictibles telle qu'à chaque prédiction locale, une unique structure est associée et que l'ensemble des prédictions locales couvre la séquence entière.

Plus les structures composant une telle librairie sont de petite taille, plus la prédiction sera facilement généralisable à de nouvelles séquences. Dans un même temps, plus les structures de la librairie sont petites, moins la cohérence de la détection sera assurée (un petit polypeptide aura plus de chance d'avoir sa structure modifiée par l'environnement structural qu'un domaine entier par exemple). C'est pourquoi il doit exister un compromis entre la couverture d'une telle librairie, son nombre d'éléments structuraux et la taille de chacun de ses éléments. On peut par ailleurs se demander quel type de structure composant la librairie permettent de minimiser sa taille ainsi que de maximiser son potentiel de généralisation à de nouvelles séquences : fragments de taille fixe ? de quelle taille ? des paires de fragments ? des motifs non-contigus ?

Par ailleurs, on sait que des protéines possèdent des fragments intrinsèquement désordonnés, et qu'il est donc impossible de prédire une structure sur ces segments de séquence. Il serait alors intéressant de définir une notion de *couverture partielle* se

focalisant par exemple sur la prédiction du cœur structural plutôt que sur la structure complète.

Désambiguïsation de prédictions locales On peut remarquer de plus que pour réaliser la prédiction d'une structure depuis la séquence, il n'est pas toujours nécessaire d'avoir des modèles de séquences cohérents : considérons une séquence q de protéine sont la structure est à déterminer. Si deux modèles (voir figure 4.4) M_{t_a} et M_{t_b} reconnaissent une même sous-chaîne q_m de q, l'ambiguïté de choix entre la structure a et b au niveau de q_m peut être résolue par la reconnaissance d'une structure c par un modèle M_c au niveau d'une sous-chaîne de q se chevauchant à q_m.

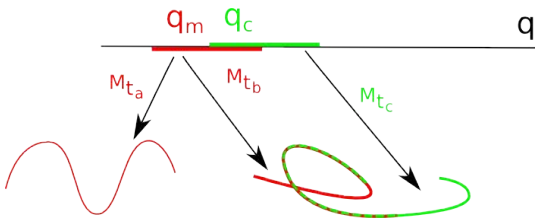

FIG. 4.4: Désambiguïsation de prédictions locales par chevauchement de prédictions compatibles.

On voit ainsi que la cohérence locale de chaque modèle n'est pas nécessaire, mais que les levées d'ambiguïtés par chevauchement peuvent provoquer l'unicité de la prédiction globale de structure.

On présente ici une perspective de développement du cadre formel de la section précédente permettant de capturer simultanément la notion de fragments, d'échelle et d'assemblage.

Les séquences et les structures s'organisent hiérarchiquement : la relation $q_a \leq q_b \Leftrightarrow$ (q_a est une sous-chaîne de q_b) est une relation d'ordre partiel entre les séquences. On peut établir de la même manière un ordre partiel sur l'ensemble des structures : $t_a \leq t_b \Leftrightarrow$ (t_a est une sous-structure de t_b)[2].

On peut de plus définir une structure de catégorie [Mac78] sur l'ensemble des fragments de structures :

- Un objet est un ensemble de fragments de même longueur
- Un morphisme $f : X \to Y$ est une fonction telle que $\forall x \in X, f(x) \leq x$, où $\leq$ est la relation précédemment définie d'ordre partiel entre structures.

En considérant l'ensemble partiellement ordonné des séquences comme une catégorie, l'application $P : q \mapsto \{q\}'$ est un préfaisceau [Mac78], c'est-à-dire que pour tout $q_a \leq q_b \in \Omega_q$, $\forall t_b \in P(q_b), \exists t_a \in P(q_a)$ tel que $t_a \leq t_b$.

On pourrait alors redéfinir dans ce formalisme une *couverture* de l'ensemble des protéines comme étant un *site* sur la catégorie des séquences tel que le préfaisceau

[2]On dira que $b_1, ..., b_M \in \mathbb{R}^3$ est une sous-structure d'une structure $a_1, ..., a_N \in \mathbb{R}^3$ s'il existe un entier $1 \leq i \leq N - M$ tel que $a_{i+1}, ...a_{i+M}$ est superposable à $b_1, ..., b_M$.

P soit un faisceau. Cette définition est relativement large car elle n'impose pas que chaque fragment utilisé dans la couverture soit faiblement prédictible, mais seulement que l'assemblage de ceux-ci soit unique.

Dans cette thèse nous proposons des avancées dans cette direction en introduisant les fragments en contact (chapitre 6) dont nous évaluerons (section 6.4) le gain de cohérence par rapport à des fragments simples ou des paires de fragments qui ne seraient pas en contact. La notion de prédictibilité que nous avons définie nécessite tout d'abord de munir l'espace des structures d'une mesure de dissimilarité d entre les fragments. Dans le prochain chapitre, nous introduirons l'ASD, une nouvelle mesure de dissimilarité structurale permettant la comparaison non seulement de fragments simples mais aussi des fragments en contact.

Chapitre 5

ASD : comparaison de la divergence globale de fragments de structures

Les structures locales dans les protéines sont importantes pour plusieurs raisons. D'une part les conservations de séquence et de structure ne sont souvent pas uniformes et la pression de sélection est essentiellement exercée sur des fragments essentiels à la fonction de la protéine : maintien global de la structure via des interactions dites *enfouies* (comme dans le cas des *core fold* par exemple), conservation de la séquence et de la structure d'un site catalytique, etc. D'autre part, d'un point de vue informatique, les fragments peuvent constituer des briques de base pour la description (type ensemble de fragments) ou la prédiction de structure (via alphabets structuraux par exemple).

Dans toutes les tâches bioinformatiques impliquant l'utilisation de fragments de structures, un des points essentiels est leur comparaison. Nous avons déjà vu qu'il existe de nombreuses méthodes pour cela, dont la plus courante est le RMSD. Les désavantages de ce dernier sont bien connus : la valeur tend à augmenter avec la longueur des fragments, les variations locales sont prédominantes sur la similarité globale (une forte déviation de la position d'un seul acide-aminé peut donner un RMSD total relativement mauvais même si le reste du fragment est totalement superposé). D'autres scores ont plus récemment émergé pour la comparaison de fragments : le TM-score [ZS04] et le BC score [GT13]. Ces deux scores sont normalisés et ne voient pas leur distribution dépendre de la longueur des fragments. De plus, le TM-score limite la participation de chaque acide-aminé, et évite ainsi qu'un seul résidu dégrade le score alors que l'ensemble de la structure à comparer est similaire. Dans un autre registre, le BC score permet d'identifier une structure et son miroir.

Pour tous les scores précédemment cités, on suppose qu'un alignement des acides-aminés est réalisé entre les structures à comparer. Ou bien cet alignement est tacite et les acides-aminés sont alignés un-à-un (*i.e.* l'acide-aminé en position k de la première structure est aligné avec l'acide-aminé k de la seconde, pour tout k), ou bien cet alignement résulte d'une optimisation du score. Dans le premier cas, s'il existe des indels, la comparaison de structure n'aura que peu de sens car les acides-aminés comparés seront décalés. Aussi, dans le second cas, les indels sont correctement gérés puisque on autorise

généralement l'ouverture de gap dans l'alignement. Bien souvent ce sont des heuristiques qui permettent de générer l'alignement en optimisant au mieux le score de similarité de la structure (comme TM-align pour le TM-score). Ces heuristiques ne garantissent pas un score optimal comme le ferait un algorithme exact (formule de Kabsch [Kab76] pour le RMSD par exemple). Il existe de plus nécessairement un compromis entre la longueur de l'alignement et la qualité de l'alignement. Ce compromis peut être explicite (l'utilisateur peut choisir un ratio RMSD/longueur d'alignement par exemple), ou implicite s'il est masqué dans la normalisation du score (comme c'est le cas avec le TM-Score) où l'on maximise le score sans tenir compte de la longueur de l'alignement généré. Enfin, lorsqu'on calcule un alignement, le score de similarité résultant ne prendra pas compte des acides-aminés non-alignés. Ceci peut être pourtant essentiel pour certaines applications s'intéressant à la structure entière d'un fragment. Aussi, en n'évaluant pas la similarité sur l'ensemble des fragments la situation suivante peut se produire : une protéine A se superpose à une protéine B sur les premiers 50% de sa structure, et une protéine C se superpose à B sur les derniers 50% de sa structure, mais A ne se superpose nulle part à C. La mesure de similarité évaluant les structures seulement sur les parties alignées pourra donner un bon score entre A et B ainsi qu'entre B et C, et un score très mauvais entre A et C. Mathématiquement, on dira que la mesure globale de similarité ne respecte pas l'inégalité triangulaire. Nous dirons que *toute comparaison de structure se basant sur un alignement partiel des acides-aminés nie nécessairement l'inégalité triangulaire.* Il résultera par exemple une mauvaise classification des structures lors d'une tâche de classification automatique [Koe01].

Pour pallier à ces problèmes liés à l'alignement pour la comparaison de structures, j'ai introduit l'ASD (pour *Amplitude Spectrum Distance*), une mesure de similarité de structure basé sur la transformée de Fourier des matrices de distances internes des structures à comparer. L'ASD compare de manière globale les structures, ne nécessite ni superposition ni alignement préalable des acides-aminés, mais est néanmoins robuste aux indels, et respecte l'inégalité triangulaire. Cette dernière propriété comme déjà mentionné permet d'obtenir de meilleures classifications de structures (cf. [Koe01], ainsi que les expérimentations sur l'ASD). Par ailleurs, cette propriété permet aussi de réaliser des algorithmes plus efficaces pour la classification et l'identification de plus proches voisins [CPZ97].

L'ASD a fait l'objet d'un article publié dans BMC Bioinformatics [GC15]. Nous reprenons avec un peu plus de détails les résultats de cet article.

5.1 Définition

Nous rappelons ici que nous identifions une structure de protéine P aux points dans l'espace tri-dimensionnel Euclidien $p_0, ...p_{N-1} \in \mathbb{R}^3$ correspondant aux atomes de C_α dans l'ordre de la séquence.

On dénote D_P la matrice des distances internes de la protéine P :

$$D_{P\,i,j} := d(p_i, p_j) \tag{5.1}$$

où d est la distance euclidienne usuelle dans $\mathbb{R}^3$.

L'ASD est basée sur la transformée de Fourier, aussi nous rappelons d'abord sa définition et son contexte d'utilisation. Le lecteur intéressé peut approfondir les quelques notions présentées ci-dessous avec des ouvrages de référence comme [RKH11]. La transformée de Fourier est un opérateur permettant de décomposer un signal (une fonction continue $f : \mathbb{R} \to \mathbb{C}$) en une somme de signaux oscillatoires élémentaires ayant une fréquence propre. Il existe une correspondance exacte (bijective) entre un signal et la liste des amplitudes et phases associées à la fréquence de chaque signal élémentaire. La phase représente le décalage dans le temps de la contribution d'un signal élémentaire au signal total et l'amplitude représente l'importance de la contribution. Ainsi la transformée de Fourier du signal $x(t)$ est $X(f) := \int_{\mathbb{R}} x(t)e^{-2i\pi ft}dt$, $|X(f)|$ est l'amplitude de la contribution de fréquence f à x et $\arg X(f)$ est la phase de la contribution de fréquence f à x. On peut alors recomposer le signal par la formule de la transformée de Fourier inverse : $x(t) = \int_{\mathbb{R}} X(f)e^{2i\pi ft}df$, correspondant bien à la somme pondérée par X des différents signaux oscillatoires élémentaires $t \longmapsto e^{2i\pi ft}$.

La transformée de Fourier discrète est un cas particulier de la transformée de Fourier appliquée à des signaux discrets (*i.e.* des séquences) qu'on retrouve dans la plupart des applications traitant des signaux numériques (égaliseur et filtres de sons numériques, démodulation de signal radio dans les téléphones portables, etc.). On peut généraliser la notion de transformée de Fourier et de transformée de Fourier discrète à des signaux à deux dimensions comme les matrices (*e.g.* de images monochromes par exemple) : les signaux élémentaires sont alors des compositions de signaux oscillatoires verticaux et de signaux oscillatoires horizontaux. Dans notre cas, le signal que nous utiliserons sera la matrice D_P des distances internes.

On dénotera par $\mathcal{F}M$ la transformée de Fourier unitaire en deux-dimensions [Jai89] d'une matrice N-carrée M :

$$\mathcal{F}M_{m,n} := \frac{1}{N}\sum_{p=0}^{N-1}\sum_{q=0}^{N-1} M_{p,q}e^{-2i\pi(\frac{pm}{N}+\frac{qn}{N})} \tag{5.2}$$

$|M_{p,q}|$ représente l'amplitude de la contribution de fréquence horizontale p et de fréquence verticale q à la matrice M, et $argM_{p,q}$ représente la phase de cette contribution.

On dénote par $|M|$ la matrice dont les coefficients sont les modules des coefficients de la matrice M :

$$\forall\ 0 \leq i,j \leq N-1,\ |M|_{i,j} := |M_{i,j}| \tag{5.3}$$

On définit alors l'ASD comme la dissimilarité entre deux structures P et Q de même longueur de séquence en considérant la distance euclidienne entre les spectres des matrices de distances internes associées :

Définition 5.1 (Amplitude Spectrum Distance)**.**

$$ASD(P,Q) := ||\ |\mathcal{F}D_P| - |\mathcal{F}D_Q|\ ||_2 \tag{5.4}$$

où $||.||_2$ *est la 2-norme usuelle :*

$$||M||_2 := \sqrt{\sum_{0 \leq i,j \leq N-1} |M_{i,j}|^2} \tag{5.5}$$

La valeur exacte de l'ASD peut être calculée efficacement avec un algorithme en $O\big(N^2 \log N\big)$ [CT65].

Derrière cette définition, l'idée est que plutôt que de comparer unes-à-unes les distances entre les C_α comme dans le cas du RMSD_d , on compare des caractéristiques globales (les amplitudes des coefficients de Fourier de ces distances). En se concentrant sur les amplitudes, et en oubliant la partie "phase" du signal de ces matrices, la mesure devient invariante au décalage d'acide-aminé, et devient ainsi plus robuste aux indels. Cette intuition théorique est formalisée mathématiquement dans la section suivante, mas aussi confirmée expérimentalement dans la section 5.5.

5.2 Propriétés

Dans cette section nous montrons d'un point de vue théorique les propriétés intéressantes dont bénéficie l'ASD.

5.2.1 Propriétés essentielles pour la comparaison de structures

Invariance par isométrie On souhaite que la mesure de similarité entre structures de protéine soit indépendante du repère choisi. On appelle cela formellement l'invariance par isométrie directe (toute combinaison de translation et de rotation).

Cette propriété est relativement immédiate étant donné que l'ASD utilise uniquement les distances internes entre les atomes C_α. Ainsi $D_{F\,P} = D_P$ pour toute isométrie F, et par conséquent $ASD(F\,P, Q) = ASD(P, Q)$ pour toutes protéines P et Q.

On voit par ailleurs que cette propriété reste vraie pour les isométries indirectes et donc par symétrie (*i.e.* miroir). Ainsi, nous avons $ASD(S\,P, Q) = ASD(P, Q)$ pour toute symétrie S. Cette caractéristique est commune à tous les scores se basant sur les distances internes. Si on désire différencier une structure et son miroir, on peut le faire simplement en calculant le signe du déterminant $det(P^\top Q)$ où P et Q sont les matrices $N \times 3$ des coordonnées des atomes de C_α. En effet, un déterminant positif signifie que les structures P et Q ne sont pas miroir l'une de l'autre, alors qu'un déterminant négatif signifie que leur superposition sera meilleure en prenant le symétrique de l'une d'entre elles [GT10].

Borne Euclidienne et cohérence avec le RMSD_d Nous montrons ici une borne théorique par rapport au RMSD_d : si deux fragments sont considéré comme similaires avec le RMSD_d , alors ils seront considérés comme similaires par l'ASD.

Formellement, soit P, Q deux structures considérées comme similaires (*i.e.* on aligne les acides-aminés un-à-un). Le $\mathrm{RMSD_d}$ vaut alors :

$$\mathrm{RMSD_d}(P,Q) := \sqrt{\frac{1}{\binom{N}{2}} \sum_{i<j<n} (D_{P_{i,j}} - D_{Q_{i,j}})^2} \tag{5.6}$$

On peut alors borner l'ASD proportionnellement à $\mathrm{RMSD_d}(P,Q)$:

$$\begin{array}{rcl} \mathrm{ASD}(P,Q) & = & || \, |\mathcal{F}D_P| - |\mathcal{F}D_Q| \, ||_2 \\ & \leq & || \, \mathcal{F}D_P - \mathcal{F}D_Q \, ||_2 \\ & \leq & || \, D_P - D_Q \, ||_2 \text{ car } \mathcal{F} \text{ est une unitaire} \\ & \leq & \sqrt{\binom{N}{2}} \, RMSD_{\mathrm{d}}(P,Q) \text{ par déf. du } \mathrm{RMSD_d} \end{array} \tag{5.7}$$

Ainsi, si $RMSD_{\mathrm{d}}(P,Q) \approx 0$, alors $\mathrm{ASD}(P,Q) \approx 0$.

Faible sensibilité aux faibles changements L'ASD peut se qualifier de score graduel car si on applique une faible distorsion à une structure, il en résultera au plus un changement proportionnel dans la valeur de l'ASD.

Formellement, une faible déformation peut-être décrit mathématiquement par une fonction continue $f : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ telle que $\forall x \in \mathbb{R}^3, ||x - f(x)||_2 \leq \epsilon$.

Ainsi, pour deux structures arbitraires P, Q et pour tout faible déformation f on peut montrer que :

$$\mathrm{ASD}(P,Q) - 2N\epsilon \leq \mathrm{ASD}(f\,P,Q) \leq \mathrm{ASD}(P,Q) + 2N\epsilon \tag{5.8}$$

Démonstration. En effet, pour tout $x, y \in \mathbb{R}^3$:

$$\begin{array}{rcl} ||f(x) - f(y)||_2 & = & ||f(x) - x + x - y + y - f(y)||_2 \\ & \leq & ||f(x) - x||_2 + ||x - y||_2 + ||y - f(y)||_2 \\ & \leq & ||x - y||_2 + 2 \times \epsilon \end{array} \tag{5.9}$$

Et de la même manière : $\forall x, y \in \mathbb{R}^3 ||x - y||_2 \leq ||f(x) - f(y)||_2 + 2 \times \epsilon$

On en déduit que :

$$\begin{array}{rcl} ||D_P - D_{f\,P}||_2 & = & \sqrt{\sum_{ij} (D_{P_{ij}} - D_{f\,P_{ij}})^2} \\ & \leq & \sqrt{\sum_{ij} (2\epsilon)^2} \\ & \leq & 2N\epsilon \end{array} \tag{5.10}$$

En utilisant la borne définie dans la section précédente, on a :

$$\mathrm{ASD}(P, f\,P) \leq 2N\epsilon \tag{5.11}$$

En utilisant l'inégalité triangulaire (voir plus loin section 5.2.2) on obtient l'inégalité voulue : pour deux structures arbitraires P, Q et pour toute ϵ-déformation f on a :

$$\mathrm{ASD}(P,Q) - 2N\epsilon \leq \mathrm{ASD}(f\,P,Q) \leq \mathrm{ASD}(P,Q) + 2N\epsilon \tag{5.12}$$

□

Ainsi, dans le cas de faibles variations de structures $2N\epsilon \approx 0$ et donc $\mathrm{ASD}(f\,P, Q) \approx \mathrm{ASD}(P, Q)$.

5.2.2 Propriétés spécifiques de l'ASD

Invariance par permutation circulaire On note $D_P \gg s$ la permutation circulaire de s acides-aminés de la matrice de distance D_P de P :

$$(D_Q \gg s)_{i,j} := D_{Q_{i+s,j+s}} \tag{5.13}$$

En définissant $P \gg s$ comme la protéine P permutée circulairement par s acides-aminés (*i.e.* telle que $D_{(P \gg s)} = D_P \gg s$), on peut montrer que :

$$\mathrm{ASD}(P, P \gg s) = 0 \tag{5.14}$$

Démonstration. En effet, si on considère le cas plus simple d'un vecteur N-dimensionnel $\vec{x} := (x_0, ..., x_{N-1})$ de $\mathbb{R}^N$. On définit sa permutation circulaire $\vec{y}$ ayant les coordonnées $y_k := x_{k-s}$. On a alors :

$$\begin{aligned}
\mathcal{F}\vec{y}_n &= \frac{1}{\sqrt{N}} \sum_k y_k . e^{-2i\pi kn/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_{k-s} . e^{-2i\pi kn/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_k . e^{-2i\pi (k+s)n/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_k . e^{-2i\pi kn/N} . e^{-2i\pi sn/N} \\
&= e^{-2i\pi sn/N} . \frac{1}{\sqrt{N}} \sum_k x_k . e^{-2i\pi kn/N} \\
&= e^{-2i\pi sn/N} . \mathcal{F}\vec{x}_n
\end{aligned} \tag{5.15}$$

Ainsi,

$$\forall n, |\mathcal{F}\vec{y}_n| = |\mathcal{F}\vec{x}_n| \tag{5.16}$$

Similairement, en dimension 2 on a :

$$|\mathcal{F}M| = |\mathcal{F}(M \gg s)| \tag{5.17}$$

Par conséquent, pour une structure P :

$$\mathrm{ASD}(P, P \gg s) = 0 \tag{5.18}$$

□

Cette propriété montrera son importance quand nous introduirons la variante "0-complétée" de l'ASD.

Invariance par inversion de séquence Comme exposé dans [MG99], la similarité de structure entre deux enzymes ayant convergé vers la même fonction peut se traduire par des similarités non-séquentielles : il se peut que deux structures puissent se superposer mais les structures secondaires ne seront pas liées dans le même ordre et potentiellement dans un sens opposé. Ainsi, il peut être intéressant de pouvoir comparer des similarités de structure à "sens de séquence près". Étant donné que l'ASD compare *stricto sensu* de manière globale une séquence de points dans un espace à trois dimensions, peu importe le sens de cette séquence de points, la valeur de l'ASD sera la même. L'ASD pourrait donc s'appliquer à ce type de comparaison.

Formellement, si on note $\overline{P}$ l'inversion de la séquence de la structure $P = (p_0, ..., p_{N-1})$, *i.e.* $\overline{P} = (p_{N-1}, ..., p_0)$. La matrice de distance d'une telle structure inversée en séquence est donnée par $D_{\overline{P}i,j} = D_{PN-1-i,N-1-j}$. On montre alors :

$$\mathrm{ASD}(\overline{P}, P) = 0 \tag{5.19}$$

Et pour deux structures arbitraires P, Q, on obtient de la même manière :

$$\mathrm{ASD}(\overline{P}, Q) = \mathrm{ASD}(P, Q) \tag{5.20}$$

Démonstration. Comme dans le paragraphe précédent, afin de simplifier la démonstration, considérons le cas unidimensionel d'un vecteur à N dimensions $\vec{x} := (x_0, ..., x_{N-1})$ of $\mathbb{R}^N$. On note l'inversion séquentielle du vecteur $\vec{x}$ par $\overleftarrow{x} := (x_{N-1}, ..., x_0)$, i.e. $\overleftarrow{x}_i = \vec{x}_{N-1-i}$. Alors,

$$\begin{aligned}
\mathcal{F}\overleftarrow{x}_n &= \frac{1}{\sqrt{N}} \sum_k x_{N-k}.e^{-2i\pi kn/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_k.e^{-2i\pi(N-k)n/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_k.e^{-2i\pi N/N}.e^{2i\pi kn/N} \\
&= \frac{1}{\sqrt{N}} \sum_k x_k.e^{2i\pi kn/N} \\
&= \overline{\mathcal{F}\vec{x}_n}
\end{aligned} \tag{5.21}$$

Ainsi,

$$\forall n, |\mathcal{F}\overleftarrow{x}_n| = |\mathcal{F}\vec{x}_n| \tag{5.22}$$

De la même manière on obtient le même résultat en dimension 2 : si M' est l'inversion séquentielle de la matrice M, *i.e.* $M'_{i,j} := M_{N-1-i,N-1-j}$, alors on a :

$$|\mathcal{F}M'| = |\mathcal{F}M| \tag{5.23}$$

La matrice de distance $D_{\overline{P}}$ de la structure séquentiellement inversée de P est l'inversion séquentielle de la matrice D_P. Par la précédente équation on obtient :

$$\mathrm{ASD}(\overline{P}, P) = 0 \tag{5.24}$$

Et donc pour deux structures arbitraires P, Q, en utilisant l'inégalité triangulaire :

$$\mathrm{ASD}(\overline{P}, Q) = \mathrm{ASD}(P, Q) \tag{5.25}$$

□

Cette propriété permet de d'identifier des structures ayant la même conformation sans tenir compte du sens de la séquence sous-jacente. Cette propriété n'apparaît que dans les aligneurs non-séquentiels comme MICAN [MSC13] qui sont lourds en calcul. Nous verrons plus loin quelques exemples de structures retrouvées grâce à l'ASD ayant la même conformation mais un sens de séquence inversé.

ASD est une pseudométrique Il est très intéressant pour une mesure de (dis)similarité d'être une pseudo-métrique. En effet, cela permet l'utilisation d'algorithmes efficaces étant donné que l'inégalité triangulaire est souvent requise pour réduire l'espace de recherche dans des algorithme de recherche de plus proches voisins par exemple [CPZ97].

Si on considère trois structures arbitraires P,Q,R, on peut montrer que :
- $\forall P, \mathrm{ASD}(P, P) = 0$
- $\forall P, Q, \mathrm{ASD}(P, Q) = \mathrm{ASD}(Q, P)$
- $\forall P, Q, R$,
  $\mathrm{ASD}(P, R) \leq \mathrm{ASD}(P, Q) + \mathrm{ASD}(Q, R)$

Ainsi,

Proposition 5.2.1. *ASD est une pseudo-métrique.*

Démonstration. Seule l'inégalité triangulaire mérite démonstration :

$$\begin{array}{rcl} \mathrm{ASD}(P, R) & = & || \, |\mathcal{F}D_P| - |\mathcal{F}D_R| \, ||_2 \\ & = & || \, |\mathcal{F}D_P| - |\mathcal{F}D_Q| + |\mathcal{F}D_Q| - |\mathcal{F}D_R| \, ||_2 \\ & \leq & || \, |\mathcal{F}D_P| - |\mathcal{F}D_Q| \, ||_2 + || \, |\mathcal{F}D_Q| - |\mathcal{F}D_R| \, ||_2 \\ & \leq & \mathrm{ASD}(P, Q) + \mathrm{ASD}(Q, R) \end{array} \tag{5.26}$$

Les autres propriétés étant immédiates. □

ASD n'est pas une métrique proprement dite car la condition $ASD(P, Q) = 0 \Rightarrow P = Q$ n'est pas respectée. En effet, en prenant par exemple P et Q comme étant miroir l'une de l'autre, on obtient que $ASD(P, Q) = 0$, mais $P \neq Q$).

5.3 Variantes de l'ASD

5.3.1 Variante 0-complétée

Cette version dite 0-complétée de l'ASD permet d'une part de pouvoir comparer des fragments de tailles différentes, et d'autre part de garantir une borne théorique sur la comparaison de fragments similaires à décalage de séquence près. Voyons dans un premier temps sa définition. Nous noterons $\widetilde{\mathrm{ASD}}$ cette variante dont la définition

formelle remplace les matrices de distances internes par des matrices 0-complétées $\widetilde{D_P}$ et $\widetilde{D_Q}$:

$$\widetilde{\text{ASD}}(P,Q) := || \, |\mathcal{F}\widetilde{D_P}| - |\mathcal{F}\widetilde{D_Q}| \, ||_2 \tag{5.27}$$

$\widetilde{D_P}$ et $\widetilde{D_Q}$ sont des versions (toutes deux de dimensions $N \times N$ avec $N = 2 \times \max(N_P, N_Q)$) des matrices D_P et D_Q (de dimensions respectives N_P et N_Q) entourées de 0.

On peut alors établir une borne théorique de l'$\widetilde{\text{ASD}}$ entre deux structures P and Q telles qu'elles se superposent exactement sur une sous-partie conséquente comme dans l'exemple de la figure ci-dessous :

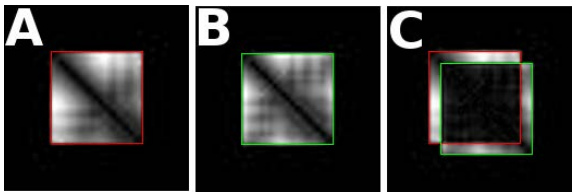

FIG. 5.1: Représentation graphique de matrices de distances internes, plus un pixel est foncé, plus la distance $d_{i,j}$ correspondante est faible. **A** : Matrice de distance 0-complétée pour les fragments 34 :54 du domaine Astral d1amk_ _ ; **B** : Matrice de distance 0-complétée pour les fragments 37 :57 du même domaine Astral ; **C** : Différence entre les deux matrices dans leur alignement optimal

On peut alors montrer que :

$$\widetilde{\text{ASD}}(P,Q) \leq ||D_{P \setminus Q}||_2 \tag{5.28}$$

où $D_{P \setminus Q}$ est la différence des matrices de distance dans un alignement optimal sans indel des matrices de distances internes comme illustré sur la figure 5.1C. Ceci signifie que *au plus* $\widetilde{\text{ASD}}$ mesure là où P et Q diffèrent.

Démonstration. Dans un premier temps supposons que nous disposons de deux structures P et Q qui se superposent parfaitement sur une sous-partie contiguë. Formellement, cela signifie qu'il existe a, b et un décalage s tels que $D_{Pi,j} = D_{Q_{i+s,j+s}} \; \forall i, j$ tels que $a \leq i, j \leq b$.

On rappelle qu'on note $D_Q \gg s$ la permutation circulaire de s acides-aminés de D_Q :

$$(D_Q \gg s)_{i,j} := D_{Q_{i+s,j+s}} \tag{5.29}$$

En utilisant l'invariance par permutation circulaire de la section 5.2.2 et la borne Euclidienne de la section 5.2.1, on obtient que :

$$\begin{array}{rcl} \text{ASD}(P,Q) & := & || \, |\mathcal{F}D_P| - |\mathcal{F}D_Q| \, ||_2 \\ & = & || \, |\mathcal{F}D_P| - |\mathcal{F}(D_Q \gg s)| \, ||_2 \\ & \leq & || \, D_P - D_Q \gg s \, ||_2 \end{array} \tag{5.30}$$

Or, $D_P - D_Q \gg s$ vaut zéro partout où P et Q sont superposés. Ainsi, l'ASD mesure au plus la différence entre P et Q seulement là où ils diffèrent.

Cette mesure peut être dénuée de sens car en permutant circulairement D_Q on compare des parties des structures P et Q sans relations (*i.e.* le début de la structure P avec la fin de la structure Q et inversement). Afin d'établir une borne plus censée, on introduit la version 0-complétée.

Considérons maintenant que nous complétons nos matrices de distances D_P (de dimension $N_P \times N_P$) et D_Q (de dimension $N_Q \times N_Q$) avec des zéros pour en faire des matrices $\widetilde{D_P}$ and $\widetilde{D_Q}$ de dimension $N \times N$ avec $N = 2 \times \max(N_P, N_Q)$.

Comme dans l'équation 5.30, on obtient :

$$\widetilde{\text{ASD}}(P,Q) \leq || \widetilde{D_P} - \widetilde{D_Q} \gg s ||_2 \tag{5.31}$$

Mais cette fois-ci $\widetilde{D_P} - \widetilde{D_Q} \gg s$ est non nul seulement où P et Q diffèrent : c'est-à-dire hors du rectangle défini par a, b. En prenant s comme étant le décalage optimal pour al superposition de P et Q, on a au final :

$$\widetilde{\text{ASD}}(P,Q) \leq ||D_{P\setminus Q}||_2 \tag{5.32}$$

où $D_{P\setminus Q}$ est la différence des matrices de distances internes dans l'alignement optimal sans indel de ces matrices (*i.e.* ayant un décalage minimisant le $RMSD_d$).

□

Etant donné que $\widetilde{ASD}$ facilite l'utilisation que la définition originale de l'ASD (car permettant notamment la comparaison de fragments de tailles différentes) et respecte exactement les mêmes propriétés que précédemment, dans la suite de ce document nous noterons ASD cette version 0-complétée avec N valant deux fois la taille maximale des fragments considérés.

5.3.2 ASD normalisée

Comme montré sur la figure 5.2A, la distribution des valeurs de l'ASD dépend largement de la longueur des fragments considérés. Pour cela j'ai introduit une version normalisée de l'ASD, notée NASD (pour *Normalized ASD*).

On définit NASD entre deux structures P et Q comme étant la pseudométrique suivante :

$$\text{NASD}(P,Q) := || \frac{|\mathcal{F}D_P|}{||D_P||_2} - \frac{|\mathcal{F}D_Q|}{||D_Q||_2} ||_2 \tag{5.33}$$

NASD réussit à normaliser les scores respectivement à la longueur des structures considérées comme on peut l'observer sur la figure 5.2B. Ceci vient au prix d'une perte d'information substantielle car on normalise *a priori* les matrices de distances internes. En effet, l'ASD présente de meilleurs résultats en pratique que NASD comme nous le verrons dans l'expérience sur la détection de fragment de Zinc Finger présenté plus loin. On observe notamment une corrélation de Pearson de 0.53 entre les valeurs de l'ASD et de NASD sur des fragments de strucure de taille 20, mais ce coefficient augmente à 0.9

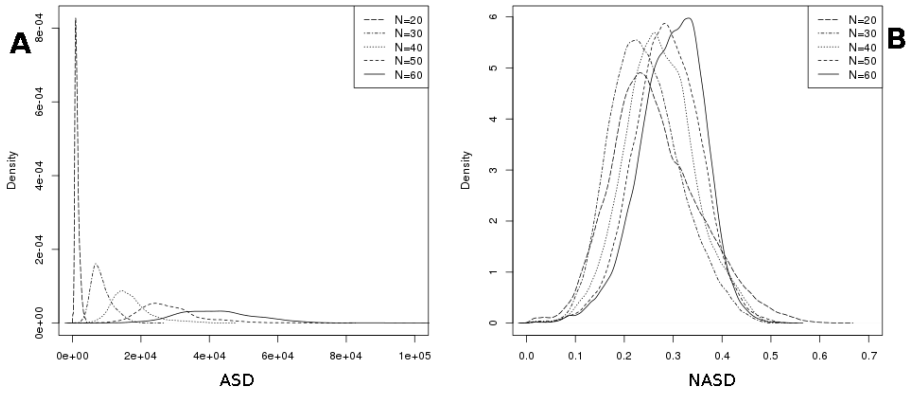

FIG. 5.2: Densité de probabilité empirique (sur les jeux de données $\mathbf{Sk}_N$ de l'ASD (**A**) et de NASD (**B**) pour des longueurs de fragments de structure de N=20, 30, 40, 50 et 60 acides-aminés.

en ne considérant que les petites valeurs de l'ASD (en dessous de 1000 dans ce cas-ci). Ces résultats sont illustrés sur la figure ci-dessous :

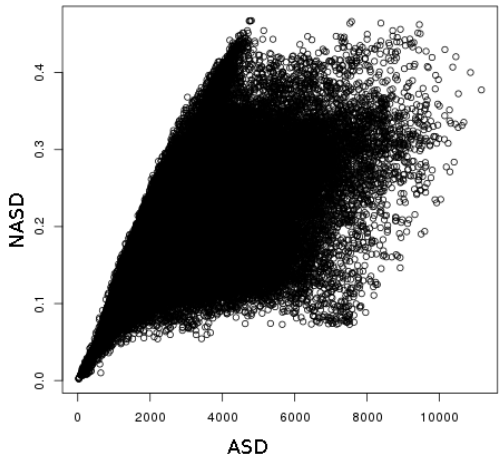

FIG. 5.3: ASD vs NASD sur toutes les paires de fragments du jeu de données $\mathbf{SkF}_{20}$, corrélation de Pearson de 0.53, mais s'élevant à 0.9 en ne considérant que les valeurs de l'ASD en-dessous de 1000

5.3.3 ASD tronquée

Pour le calcul de l'ASD, on utilise la 2-norme sur la matrice formée par les modules des coefficients de Fourier de la matrice des distances internes. Ainsi, on a besoin de calculer tous les coefficients de Fourier et de les comparer un-à-un.

Pour des tâches nécessitant une plus grande rapidité de calcul, on peut se contenter

de calculer seulement certains coefficients de Fourier et de comparer seulement ceux-ci pour obtenir une valeur approximative de l'ASD.

On observe que les premiers coefficients (correspondant aux "basses fréquences") ont les valeurs les plus élevées. Comme on peut le voir sur la figure 5.4, en ne gardant que les 5 premiers coefficients sur les 40×40 (soit une réduction de plus de 98%), on conserve une très bonne corrélation avec l'ASD standard (corrélation de Pearson de 0.95).

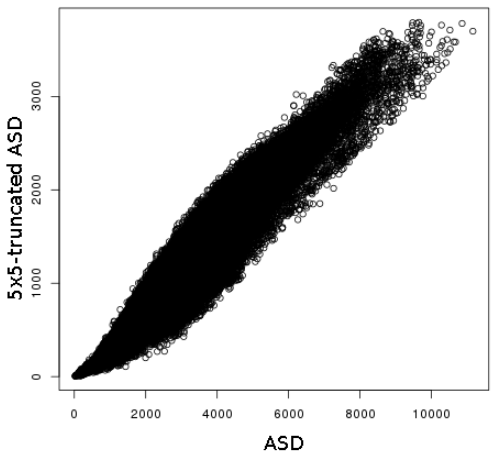

FIG. 5.4: ASD calculé sur toutes les paires de fragments du jeu de données $\mathbf{SkF}_{20}$ en considérant tous les 40×40 coefficients de Fourier vs. variante tronquée de l'ASD calculée en utilisant seulement les 5×5 premiers coefficients de Fourier, la corrélation de Pearson vaut 0.95.

5.4 Distribution de l'ASD

5.4.1 Significativité de l'ASD

Comme le RMSD, la valeur de l'ASD dépend de la longueur des fragments à comparer. D'après les comparaisons des fragments au sein des jeux de données $\mathbf{SkF}_N$, nous avons pu obtenir un seuil empirique d'ASD indiquant une similarité significative en fonction de la longueur des fragments. Ces seuils sont reportés sur la figure suivante :

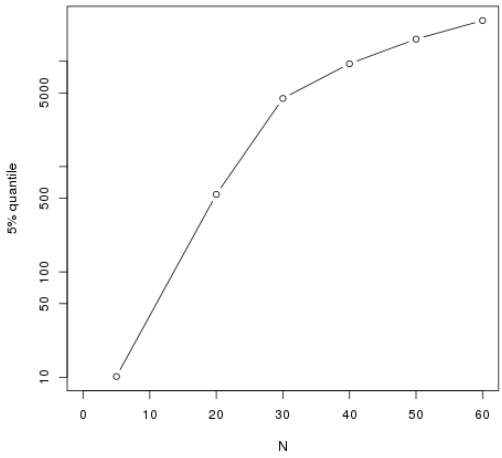

FIG. 5.5: Valeur du 5% quantile de la distribution de l'ASD lors de la comparaison des fragments de $\mathbf{SkF}_N$, en fonction de N.

Par exemple, pour de fragments de taille 23 le 5% quantile de la distribution est de l'ordre de 1700, *i.e.* lors de la comparaison de fragments de taille 23, si l'ASD montre une valeur inférieure à 1700, alors on peut considérer les fragments comme significativement ressemblants.

5.4.2 Distribution de l'ASD face aux autres scores

Nous montrons ici que sur les cas non-ambigus (*i.e.* entre des fragments s'alignant complètement par superposition) l'ASD est cohérente avec quelques uns des scores standard décrits en introduction : RMSD, TM-score et BC score. Nous détaillerons aussi les cas où l'ASD se distingue de ces scores standards.

Nous avons réalisé une comparaison de toutes les paires de structures du jeu de données $\mathbf{SkF}_{20}$ — soit 15,026,162 comparaisons structurales — en utilisant le RMSD, TM-score et l'ASD.

Comparaison avec le RMSD Sur la figure 5.6A on peut voir la distribution du RMSD vs. ASD. L'ASD étale la distribution du RMSD au niveau des valeurs intermédaires, mais comme on peut le voir sur la figure 5.6B, l'ASD conserve une très bonne corrélation avec le RMSD pour les fragments très similaires (*i.e.* signifiant ici entièrement superposables par l'outil TM-Align).

En analysant les exceptions (*i.e.* les points hors du cœur de la distribution) dans la distribution ASD vs RMSD, on aboutit à différentes raisons pour que l'ASD donne une bonne similarité alors que le RMSD en évalue une mauvaise :

– les deux structures ont une similarité globale, mais présentent des variations locales, empêchant notamment une bonne superposition des structures (voir figure

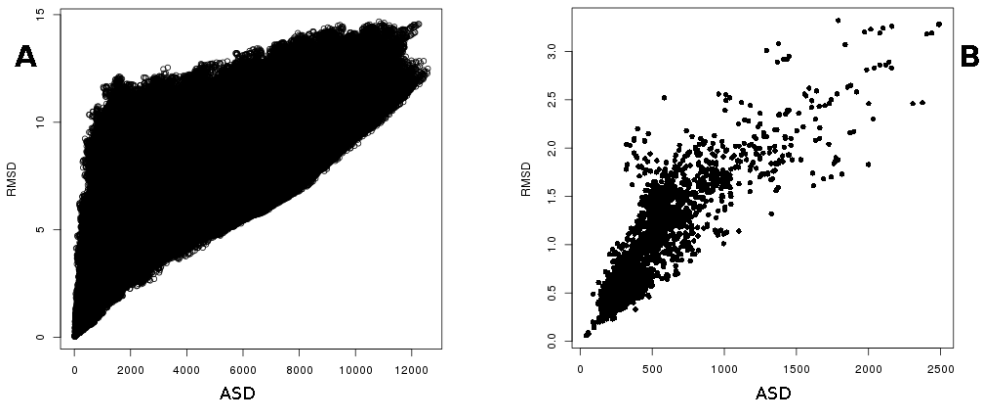

FIG. 5.6: Illustration de la corrélation entre l'ASD et le RMSD. **A** : Distribution du RMSD versus ASD sur le jeu de données $\mathbf{SkF}_{20}$, le coefficient de corrélation de Pearson est de 0.65 ; **B** : Correlation entre l'ASD et le RMSD sur le sous-ensemble des fragments de structure du jeu $\mathbf{SkF}_{20}$ entièrement alignés par l'outil Fr-TM-Align . Le coefficient de corrélation de Pearson remonte à 0.85.

5.8C pour un exemple),
- les structures sont superposables, mais en sens opposé (N-terminal et C-terminal inversés) comme l'illustre la figure 5.7,
- les structures sont miroir l'une de l'autre, conséquence de l'utilisation des distances internes,
- les structures pourraient être correctement superposées sur une partie conséquente de leur structure à un décalage de séquence près. Voir figure 5.8A pour un exemple concret entre deux fragments provenant de deux protéines différentes et figure 5.8B montrant un exemple de décalage de séquence de 4 acides-aminés dans une épingle à cheveux β.

Comparaison avec le TM-score Les exceptions dans la distribution ASD vs TM-score sont plus difficiles à classifier. Les exemples les plus frappants sont des structures qui se superposent, mais en sens inverse de séquence d'acides-aminés. Les autres différences se situent principalement au niveau des scores intermédiaires où le TM-score acte de similarités locales, alors que l'ASD juge les similarités globales. Par exemple, comme on peut le voir sur la figure 5.9, l'ASD attribue une bien meilleure similarité (et inversement pour le TM-score) à deux hélices α qui sont globalement similaires mais dont les C_α possèdent des variations locales (figure 5.9A) qu'à deux structures localement similaires mais dont la forme globale est différente (figure 5.9B).

L'ASD est cohérente avec les scores standards sur les cas non-ambigus et se distingue par sa mesure davantage axée sur la similarité globale des fragments plutôt que sur la mesure des variations locales de structure. Elle permet de reconnaître la similarité

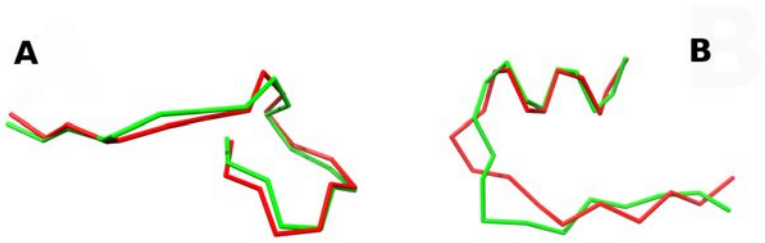

FIG. 5.7: Exemples de structures du jeu de données SkF_{20} se superposant correctement en sens inverse de séquence (*i.e.* en inversant N-terminal et C-terminal). **A** : Fragments (Astral d1b9bA, positions 60 :80, en rouge, et d1treA positions 60 :80 en vert), avec une très bonne similarité du point de vue de l'ASD (358) et un mauvais TM-score (0.38), alors que les structures se superposent quasiment en alignant le C-terminal de l'une au N-terminal de l'autre. **B** : Fragments (Astral d1qmpD, positions 74 :94, en vert, et d1qmpC positions 63 :83, en rouge) ayant une valeur d'ASD correcte (797) mais un très mauvais TM-score (0.11) alors que les structures ont une similarité de structure globale avec de faibles variations locales en les considérant en sens inverse de séquence l'une de l'autre.

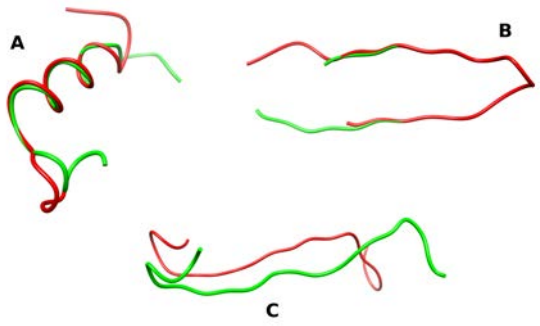

FIG. 5.8: Différences entre le RMSD (fragments considérés comme similaire par l'ASD et différents par le RMSD) : les scores sont calculés sur les fragments de structrues et l'illustration donne le meilleur alignement que l'on a pu obtenir manuellement. **A** : Astral d2lvfa, positions 95 :117 (rouge) et Astral d3ruac, positions 48 :70 (vert), le RMSD est de 4.8 et l'ASD de 1446.7 ; **B** : Astral d1x7sa, positions 78 :100 (rouge) et positions 82 :104 (vert) le RMSD est de 9.3Å et l'ASD de 1472.3 ; **C** : Astral d3rufa, positions 180 :202 (rouge) et Astral d3w29a, positions 239 :261 (vert), le RMSD est de 4.9Å et l'ASD de 1357.8.

entre fragments non-alignés, et tolère les décalages de séquence. Nous allons montrer maintenant des expériences validant sa robustesse aux indels ainsi que ses qualités pour la classification de fragments de structure.

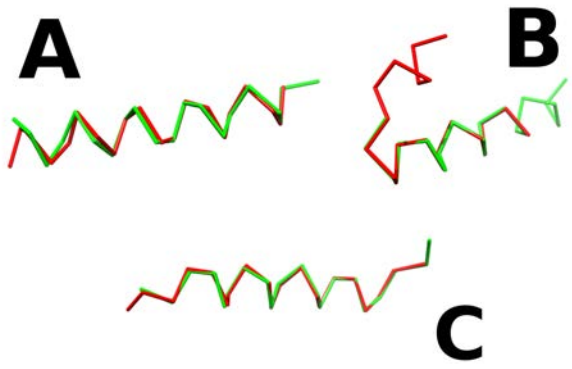

FIG. 5.9: Différence avec le TM-score : les scores sont calculés sur les fragments structuraux entiers et l'illustration montre un alignement manuel. **A :**Astral d1b71A_ positions 124 :144 (rouge) et d1psA_ positions 113 :133 (vert) sont considérés similaires par l'ASD (562) mais pas par le TM-score (0.44). **B :** Astral d1fha__ positions 109 :129 (rouge) et d1rcd_ positions 102 :122 (vert) sont similiaires pour le TM-score (0.60) mais pas pour l'ASD (2995), dans l'alignement manuel, 14 acides-aminés sur 20 ont été alignés. **C :** Astral d1amk_, positions 136 :156, en rouge, et d1tri_ positions 126 :146 sont des exemples de fragments de structures considérés comme similaires par l'ASD et le TM-score mais dont l'heuristique de Fr-TM-align donne de mauvais résultat : le TM-score entre les deux fragments est de 0.7, alors que Fr-TM-align donne un TM-score de 0.27 en alignant uniquement 11 acides-aminés.

5.5 Expérimentations

Nous présentons ici plusieurs applications que l'on peut faire de l'ASD. La première a pour but de montrer que l'ASD est plus robuste aux indels que les mesures de similarités concurrentes. En effet, l'expérience consiste à retrouver les instances d'un motif structural de Zinc Finger dans une base de données à partir d'un unique exemple structural, appelé *graine* dans la suite. Comme le Zinc Finger est un motif comprenant plusieurs sites d'indel, les scores de similarité habituels réalisent une mesure faussée car ils se basent sur un alignement un-à-un des acides-aminés (décalage dès le premier site d'indel).

Les deux autres expériences attestent de l'intérêt d'utiliser l'ASD dans des tâches de classifications : dans les deux exemples, les clusters définis par l'ASD sont bien plus nets qu'avec les mesures de similarités existantes.

5.5.1 ZF

Un Zinc Finger (ou simplement ZF) est un motif structural protéique capable de se lier à des ions de zinc. Il existe plusieurs variantes de ces motifs, ayant chacune une caractérisation en séquence. Nous utiliserons pour cet exemple le motif C2H2 *PS00028*

de Prosite (Release 20.99 [SCC+13b]) C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H.

L'expérience réalisée ici est la suivante. Nous avons extrait les fragments de structures associés au motif *PS00028*, grâce au référencement croisé entre Prosite et la PDB. Étant donné la longueur variable que peuvent prendre les instances de ce motif, afin de pourvoir comparer l'ASD avec des scores ne comparant que des fragment de même longueur, nous avons extrait systématiquement les 321 fragments de longueur 23 (taille de la plus grande instance de *PS00028*) à partir de la première position de chaque hit du motif.

Par ailleurs, depuis le jeu de données Astral64 (voir détails en annexe), nous avons extrait systématiquement tous les fragments de longueur 23 (pour un total de 10,587 fragments qui ne sont donc pas des ZF).

Enfin, nous avons mélangé les ZF et les non-ZFs, pour obtenir un jeu de fragments qu'on qualifiera de *mixte* dans la suite.

A partir d'une graine (une instance structurale d'un ZF), le but de cette expérience est de retrouver parmi le jeu de données mixte les vrais instances de ZF. Pour cela, nous comparons (en utilisant successivement l'ASD, la NASD, le RMSD, le TM-score et le BC score) la graine avec tous les fragments du jeu de données mixte et nous trions de la meilleure similarité à la plus mauvaise. Si la plupart des ZF sont classés dans les premiers, cela signifie que la mesure de similarité utilisée est sensée, sinon, cela signifie qu'elle n'arrive pas à reconnaître précisément une instance d'un ZF d'un autre fragment (notamment à cause des insertions pouvant nuire à la comparaison). L'expérience est réalisé en mode "jack-knife" c'est-à-dire en répétant l'expérience 321 fois avec à chaque fois une graine différente parmi les ZF. Les résultats indiquent les valeurs statistiques des observations sur ces 321 essais. Étant donné qu'il y a bien plus non-ZF que de ZF dans le jeu de données mixte, nous mesurons l'aire sous la courbe de Précision-Rappel (PR) plutôt que l'aire sous la courbe ROC (Reciever Operator Characterisic) — voir annexe pour plus de détails.

Une vision concrète des courbes PR est donnée sur deux exemples typiques de ces courbes en figure 5.12. On voit sur la figure 5.10 que l'ASD a généralement la meilleure AUC-PR parmi tous les autres scores testés. Notamment l'ASD est significativement meilleure que le RMSD qui arrive en seconde position en matière d'AUC-PR (Welch t-test entre les valeurs de l'ASD et du RMSD de $1.5.10^{-10}$). De plus la figure 5.11 indique que la différence se situe notamment dans la précision pour de forts rappels (l'ASD a une précision meilleure de 26% que le RMSD à 90% de rappel). Cette excellente précision pour un fort rappel contraste avec les résultats du BC score qui semble être très spécifique de la graine en question et ne supporte visiblement pas les indels dans les structures à comparer. Cependant ce score peut être considéré comme complémentaire de l'ASD en ce sens qu'il permettrait de discriminer à un niveau plus fin entre deux types de structures fortement similaires.

Comme nous l'avons déjà mentionné, l'ASD est invariante par symétrie miroir. Cette symétrie peut se détecter facilement en calculant le signe d'un déterminant (voir section 5.2.1). Nous avons ainsi réalisé la même expérimentation en classant en premier selon l'ASD les fragments ayant un déterminant positif avec la graine, puis en second, toujours selon l'ASD les fragments ayant un déterminant négatif. Les résultats liées à cette

méthode sont présentés sous le nom "ASDasym" sur les figures 5.10 et 5.11. On voit une amélioration significative des résultats par rapport à l'ASD standard : ASDasym a une précision moyenne 44% supérieure au RMSD à 90% de rappel.

Le TM-score affiche les résultats les moins bons pour cette tâche. La figure 5.13 fournit le détail de la distribution TM-score vs ASD ainsi que des exemples concrets des faux-positifs détectés par chacun des scores. Nous avons également testé l'outil Fr-TM-align pour voir s'il améliorait les résultats du TM-score. La conclusion est mitigée : d'une part cet outil s'avère trop lent pour être utilisé massivement sur toutes les comparaisons de fragments nécessaires, et d'autre part même s'il semble pouvoir améliorer les résultats sur les quelques exemples testés, il peut aussi s'avérer néfaste et donner de plus mauvais résultats que le TM-score brut. En effet, comme on peut le voir sur les courbes PR détaillées sur la figure 5.12, Fr-TM-Align donne un PR AUC de 62% pour la graine 1BBO positions 32 :54 lorsque l'ASD atteint 98% et 83% pour le BC score. Et lorsque Fr-TM-Align donne de meilleurs résultats sur l'ensemble — avec une PR AUC de 86% pour la graine 1A1F positions 107 :130 alors qu'elle vaut 83% et 85% pour l'ASD et l'ASDasym respectivement —, la précision pour un rappel de 90% n'est seulement que de 58% alors qu'elle vaut 72% et 78% pour l'ASD et l'ASDasym respectivement.

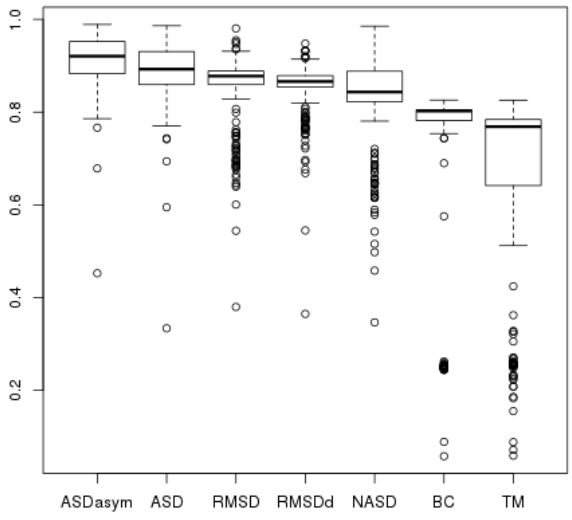

FIG. 5.10: Statistiques sur toutes les graines des PR-AUC.

5.5.2 L1-CDR

Les anticorps sont des protéines jouant un rôle essentiel dans le système immunitaire en se liant de manière spécifique à un antigène. Plus précisément seulement une petite

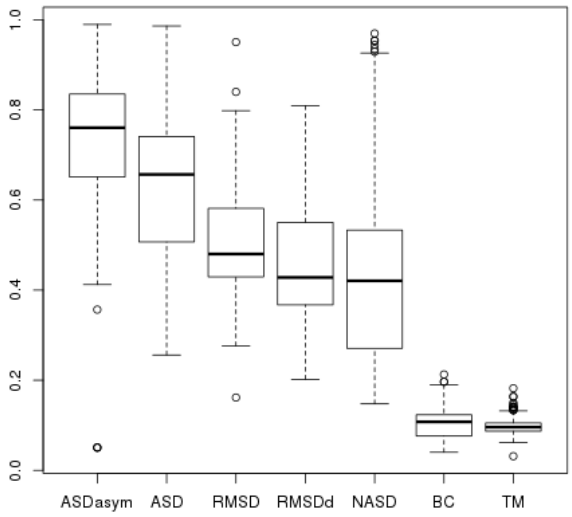

FIG. 5.11: Statistiques sur toutes les graines de la précision des scores à un rappel de 90%.

partie des anticorps se lie à l'antigène : c'est la CDR (pour *complementary-determining region*). Cette CDR est composée de six fragments notés dans littérature L1, L2, L3 (sur une chaîne dite "légère") et H1, H2 et H3 (sur une chaîne dite "lourde"). Le papier [DKL+14] décrit SAbDab, une base de données classant des structures d'anticorps selon une classification experte récente établie par [NLJ11].

On présente ici une classification automatique des fragments L1 CDR en utilisant l'ASD comme mesure de similarité. Ces fragments étant de longueur variable, les scores habituels ne peuvent pas être utilisés pour comparer ces fragments. Comme référence, nous avons comparé nos résultats à ceux pouvant être obtenus avec l'aligneur structural TM-align [ZS05]. Enfin, nous comparons nos résultats à la référence actuelle de classification [NLJ11].

La figure 5.14A et 5.14B montrent la réduction multi-dimensionnelle (MDS pour *multi-dimensional scaling*) de la dissimilarité entre les fragments telle que mesurée par l'ASD et TM-align respectivement. MDS est un moyen pratique de visualiser en basse dimension (ici, en dimension 2) les données de grande dimension. En effet, le MDS calcule la meilleure projection en deux-dimensions des données telle que deux points sont d'autant plus proches dans le plan qu'ils le sont dans leur espace d'origine. Sur les figures, les couleurs représentent les classes des fragments L1-CDR telles que définies dans [NLJ11]. La couleur grise correspondant à la classe "NA" correspond aux fragments L1-CDR n'ayant pas de classe attribuée dans la base de données SAbDab [DKL+14]. On peut voir, particulièrement en comparant à TM-Align que l'ASD forme des classes

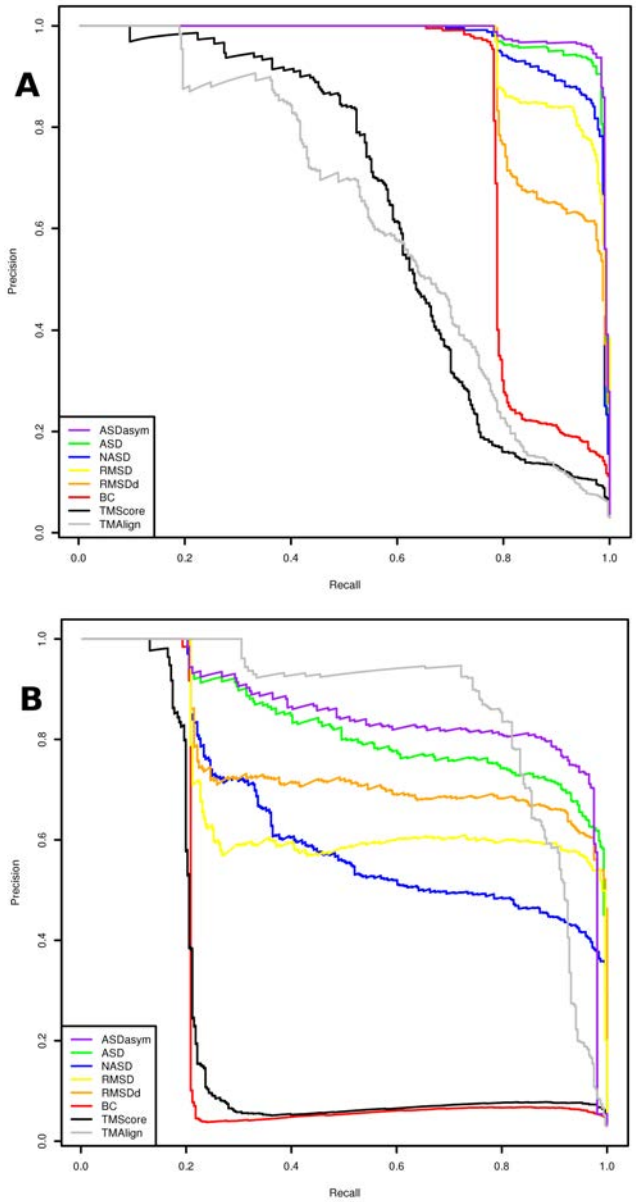

FIG. 5.12: Exemples de courbes Précision-Rappel pour chacun des scores testés dans l'expérience de détection de ZF : ASD, NASD, ASDasym, RMSD, RMSDd, BC and TM scores sont calculés sur les fragments entiers. La courbe montrant TM-Align rend compte du TM-score tel que calculé par Fr-TM-align. **A :** En utilisant le PDB 1BBO positions 32 :54 comme graine et **B :** En utilisant le PDB 1A1G positions 107 :130 comme graine.

très nettes qui correspondent dans l'ensemble aux classes définies manuellement dans

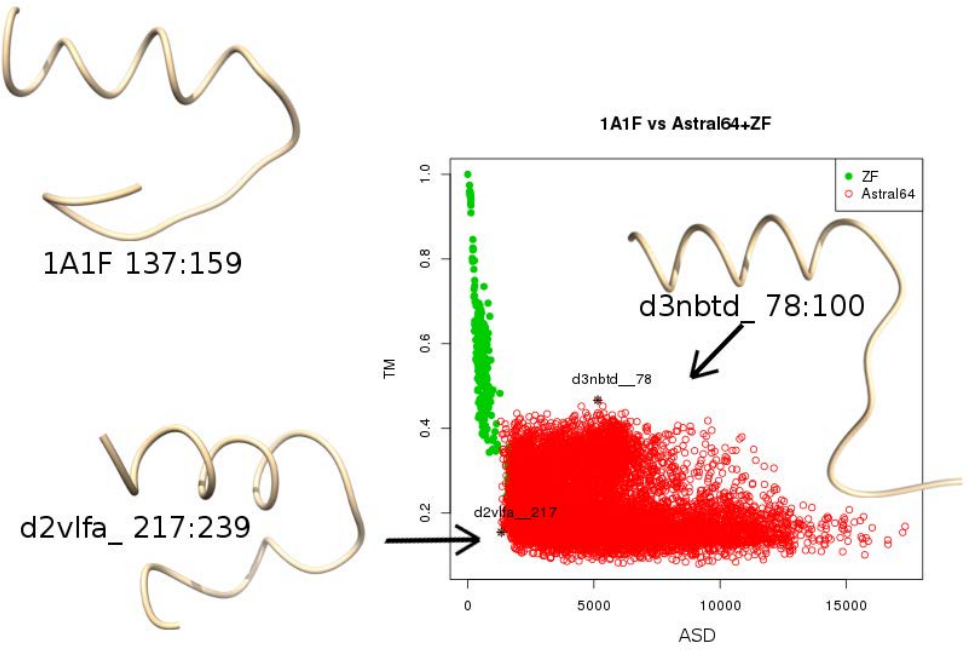

FIG. 5.13: Distribution de de l'ASD vs TM-score lors comparaison de la comparaison d'une graine (PDB 1A1F, positions 137 :159) avec les fragments de longueur 23 de Astral64 (points rouges) et fragments de ZF (points verts). La structure du fragment graine est montré en haut à gauche, ainsi que la structure de Astral d3nbtd positions 78 :100, le premier faux positif relatif au TM-score parmi les vrais négatifs de l'ASD, et le fragment Astral d2vlfa positions 217 :239, le premier faux positif de l'ASD parmi les vrais négatifs du TM-score.

[NLJ11], et que de plus, les fragments non-annotés (en gris) peuvent soit être facilement assignés à une classe existante, soit sont candidats à une nouvelle classe (comme par exemple le groupe en bas à droite sur le graphique).

Les figures 5.14C et 5.14D montrent une classification automatique hiérarchique utilisant une méthode de lien complet pour des similarités mesurées par l'ASD et TM-Align respectivement. On voit là encore que les classes sont beaucoup plus nettes avec l'ASD qu'avec TM-Align, et qu'il est donc plus facile d'attribuer une classe à un fragment en utilisant l'ASD plutôt que le TM-score tel que calculé par TM-align. Plus rigoureusement, l'index de Davies-Bouldin [DB79] mesure la qualité d'une classification dont les valeurs locales minimales peuvent être utilisées pour "couper" un dendrogramme d'une classification hiérarchique. Sur les dendrogrammes des figures 5.14C et 5.14D, nous avons trouvé des valeurs minimales de 0.2 et 0.6 pour l'ASD et TM-Align respectivement. Le DB-index étant moins élevé pour l'ASD, cela signifie que la qualité de la classification est meilleure. Les coupes du dendrogramme correspondantes mènent à la

création de 7 classes dans le cas de TM-Align avec 71.3% d'accord avec la référence de classification manuelle, alors qu'en utilisant l'ASD, on obtient 10 classes ayant un accord de 84.0% avec la référence.

Enfin, le temps de calcul a été environ 10 fois plus rapide avec l'ASD par rapport à l'utilisation de TM-align, et aurait même pu être accéléré en utilisant un algorithme de classification plus optimal profitant de la propriété d'inégalité triangulaire de l'ASD.

5.5.3 Domain linkers

Un *linker* est un fragment structural connectant deux domaines protéiques ensemble. Naturellement présents dans les protéines du vivant, les linkers sont aussi d'un intérêt particulier pour l'ingénieurie de protéines : on exprime souvent une polyprotéine unique qui possède plusieurs activités enzymatiques ; les différents domaines doivent alors être liés par des linkers ayant des caractéristiques bien précises (longueur, flexibilité par exemple). Le papier [GH02] présente une classification manuelle des linkers identifiées dans les protéines naturelles. L'attribut *hélicoïdal* vs. *non-hélicoïdal* ainsi que la longueur des linkers ont été les deux critères utilisés pour cette classification experte.

Comme dans l'expérience précédente, les fragments ayant des longueurs différentes, l'ASD peut seulement être comparée aux résultats donnés avec un outils d'alignement structural. Ici encore, nous utilisons l'outil standard TM-align. Par ailleurs, comme TM-align ne peut aligner des fragments dont la longueur est plus petite que 6 acides-aminés, la figure 5.15 montre la classification des linkers dont la longueur est comprise entre 6 et 9 acides-aminés. Nous avons également limité la longueur des fragments à 9 acides-amniés car le nombre de structures disponibles au-delà était trop maigre et ne permettait pas de trouver des classes nettes, tant avec TM-Align qu'avec l'ASD.

Comme on peut le voir sur la figure 5.15, ASD retrouve d'abord la même distinction de longueur et d'attribut hélicoïdal vs non-hélicoïdal que ceux proposés dans [GH02]. D'un autre côté, TM-align ne semble pas proposer de classes bien définies. Dans un souci de complétude, la figure 5.16 illustre la classification hiérarchique telle que définie par l'ASD pour les fragments de longueur inférieure à 6 acides-aminés.

5.6 Perspectives

Nous avons montré que dans le cas de simples fragments, l'ASD était un moyen pratique de comparer deux structures sans réaliser d'alignement préalable et tout en respectant l'inégalité triangulaire. Nous avons illustré sa capacité à réaliser des classifications claires de structures.

Plusieurs perspectives restent ouvertes. Premièrement, il faudrait creuser l'intérêt pratique des propriétés d'invariance par inversion de séquence et par symétrie miroir. Deuxièmement, nous avons évalué l'ASD sur des fragments de structures, et il pourrait être intéressant de mener à bien des expérimentations évaluant sa capacité à comparer des domaines protéiques entiers dans lesquels les indels peuvent être plus conséquents nécessitant bien souvent un alignement des structures.

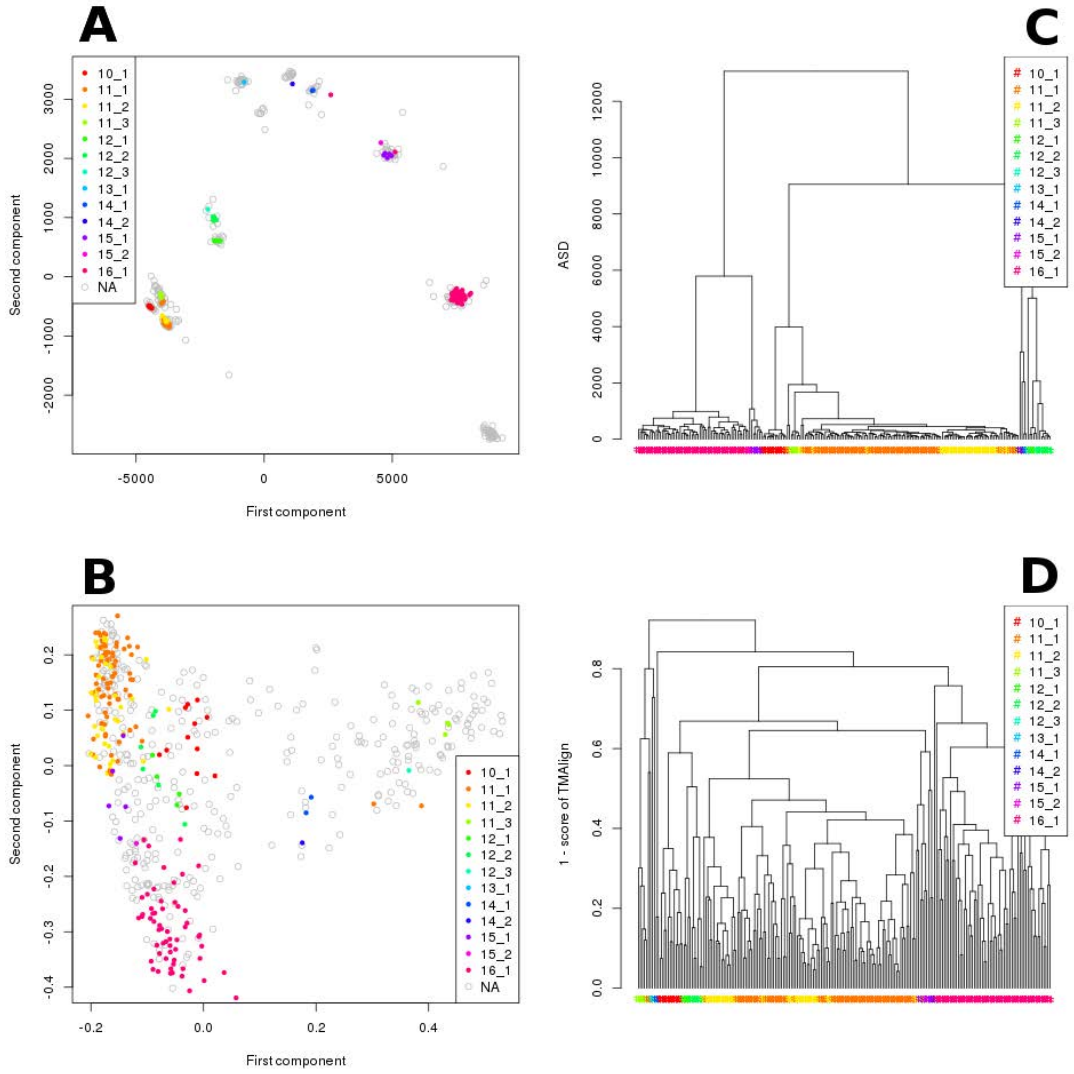

FIG. 5.14: Représentation en deux dimensions de la similarité entre les fragments L1 CDR tel que calculé avec l'ASD (A) et avec TM-Align (B). Classification hiérarchique des fragments de L1 CDR en utilisant l'ASD (C) et TM-align (D).

Nous avons introduit une variante dite tronquée de l'ASD, permettant un calcul plus rapide de la similarité en n'utilisant seulement que les premiers coefficients de Fourier des matrices de distances internes. Il pourrait être intéressant de regarder plus en détails quels sont les coefficients qui contribuent au mieux à l'évaluation de l'ASD, voire même à trouver une pondération de ces coefficients permettant la discrimination entre deux structures ou même entre deux familles de structures. On pourrait par exemple définir le spectre caractéristique d'une famille protéique.

Enfin, il est intéressant de noter que dans toute méthode utilisant la distance Euclidienne entre des vecteurs (comme par exemple l'algorithme des *k-means*), la représentation d'une structure par le *spectre* de sa matrice des distances internes permet d'obtenir directement les mêmes résultats que si on avait utilisé l'ASD comme mesure de similarité. On peut ainsi, sans aucune modification des libraires existantes, utiliser l'ASD pour faire — par exemple — du clustering.

Comme annoncé dans l'introduction générale du manuscrit, l'ASD a été initialement introduit pour comparer les fragments en contact (voir prochaine section) qui possèdent une certaine variabilité à leurs extrémités. On voulait pouvoir comparer des structures dont l'alignement optimal pouvait avoir un décalage de séquence tout en disposant d'un score qui respecterait l'inégalité triangulaire. Bien que nous ayons détaillé les propriétés de l'ASD pour le cas des simples fragments dans cette section, nous montrerons comment l'utiliser pour la comparaison de fragments en contact dans le prochain chapitre.

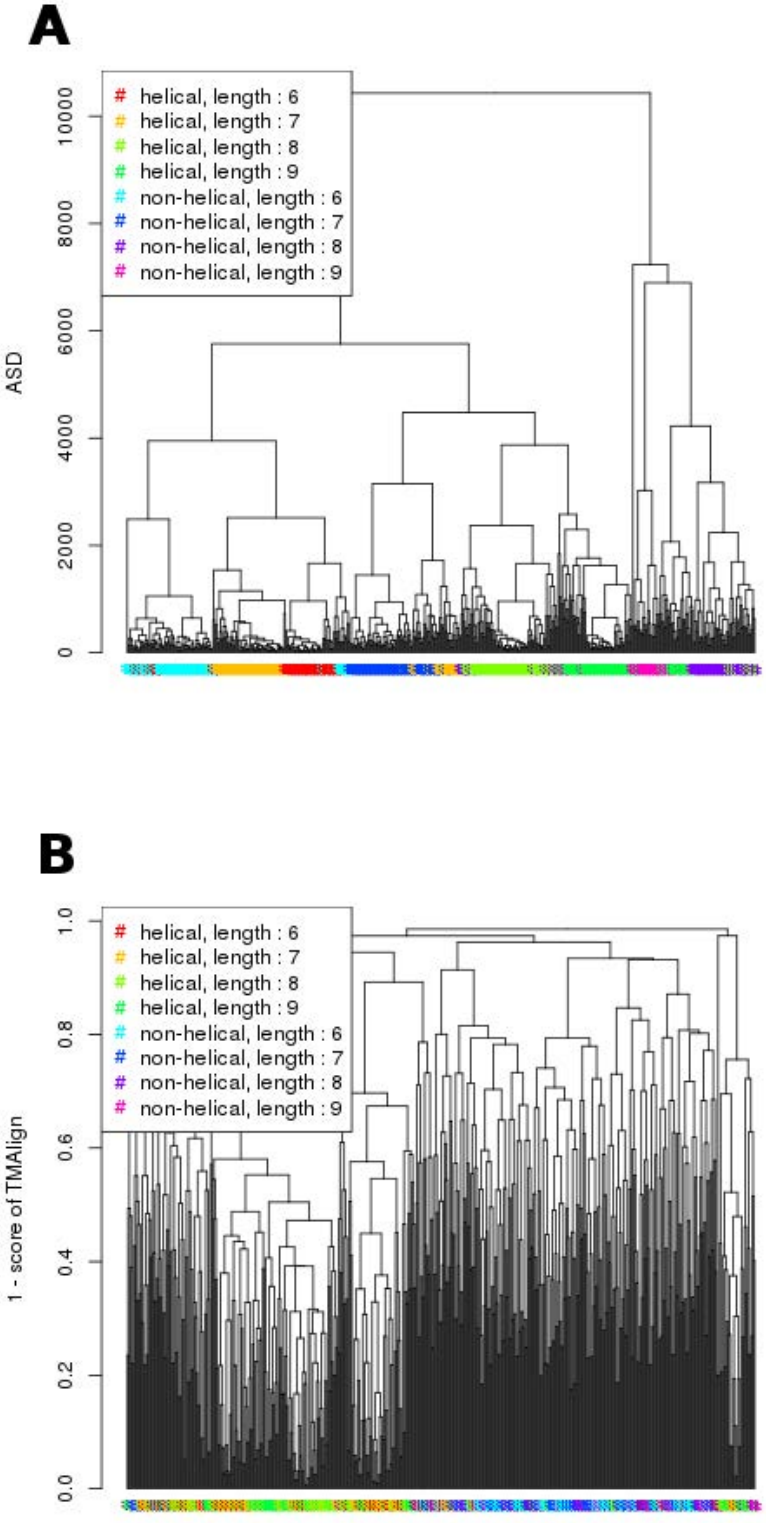

FIG. 5.15: Classification hiérarchique des linkers ayant une longueur comprise entre 6

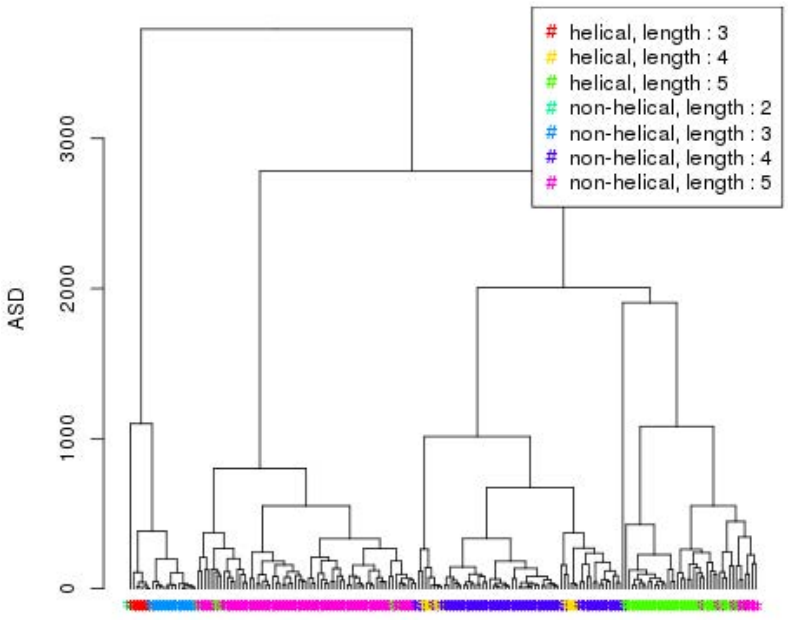

FIG. 5.16: Classification hiérarchique avec l'ASD des linkers dont la longueur est inférieure à 6 acides-aminés.

Chapitre 6

Les fragments en contact : un lieu séquence-structure privilégié

Les contacts sont fortement informatifs sur la structure des protéines. Non seulement il est possible de comparer des structures uniquement à l'aide de l'information de contact entre ses acides-aminés (voir section 2.2.0.13) mais la structure d'une protéine peut même être reconstruite uniquement avec cette information [DSS+10]. Aussi, la description d'une structure de protéine à l'aide d'une librairie de structures locales se montre plus fine lorsque la librairie comporte des paires de fragments en contact plutôt que des simples fragments contigus [XCLM13].

La structure environnante à un fragment structural influence sa conformation à travers les interactions physiques qu'elle entretient avec lui. Ceci laisse supposer que plus une portion de structure possède d'interactions internes relativement à la quantité d'interaction avec son environnement, plus elle a de chance d'être conservée dans des protéines homologues. C'est-à-dire que les séquences codant pour des structures localisées spatialement doivent donc avoir une meilleure conservation de structure que celles codant pour des structures non compactes. Selon les termes du chapitre 4, nous formulons l'hypothèse que *la localité spatiale renforce le caractère prédictible d'une structure*.

Le voisinage séquentiel des contacts semble être important à prendre en considération pour définir des signatures en séquence de familles de protéines. En effet, il a par exemple été montré dans [BRS04] que les motifs conservés qui sont en contact dans la structure peuvent être utilisés comme signatures séquentielles caractéristiques du *fold* de familles de protéines. De plus, le voisinage en séquence possède aussi un rôle dans le maintien du contact. On l'intègre notamment dans des méthodes de prédiction de contact depuis la séquence. L'outil ProfCon [PR05] prédit les contacts en utilisant une combinaison d'information locale en séquence (fréquence d'apparition d'acides-aminés dans une fenêtre autour du contact considéré, conservation dans un alignement multiple obtenu par PSI-BLAST, prédiction de la structure secondaire locale et de l'accessibilité au solvant). Également, [dLNB12] propose de prédire des interactions entre structures secondaires, puis de raffiner ces interactions pour produire des probabilités de contact au niveau des acides-aminés par une technique de minimisation d'énergie et enfin d'uti-

liser ces probabilités de contact dans une fenêtre locale de séquence afin de prédire *in fine* les contacts. C'est pourquoi, une autre hypothèse essentielle que nous faisons est que le *voisinage en séquence* appuie le caractère prédictible d'un motif structural.

Ainsi, comme nous avons pu le voir dans le chapitre 3, les motifs qui définissent des signatures caractéristiques de familles protéiques sont soit de nature séquentielle, soit de nature structurale. La séquence étant technologiquement plus accessible que la structure des protéines, il est fréquent de chercher une signature de séquence des motifs structuraux conservés. Il nous intéresse ici d'identifier des motifs structuraux resserrant le lien entre la séquence et la structure. Idéalement, on voudrait disposer d'une implication directe entre une signature en séquence et son instance structurale : chaque fois que la signature séquentielle est identifiée dans une séquence, la structure associée correspond exactement au motif structural en question. Formulé selon les termes du chapitre 4, on dira que le motif est caractéristique et *prédictible*. Un tel motif constitue alors une signature structurale et séquentielle.

Dans la suite de ce chapitre, nous introduisons les *fragments en contact* (ou CF pour *contact fragments*), qui visent à concilier localité de séquence et localité spatiale en vue d'obtenir des motifs capables de fournir des signatures structurales prédictibles. Nous verrons dans un premier temps la définition de ces CF, puis quelques unes de leurs propriétés informatiques et statistiques. On présentera ensuite une expérience montrant qu'en utilisant un modèle de séquence basé sur l'homologie, les CF sont davantage prédictibles que des fragments de séquence simples ainsi que de paires de fragments qui ne seraient pas en contact dans la structure. Enfin, nous verrons une utilisation concrète à travers l'outil VIRALpro développé dans cette thèse [GMCB15] ainsi que quelques perspectives possibles d'investigation.

6.1 Définition

Les *fragments en contact* (ou CF pour *contact fragments*) sont des portions de structure conciliant la localité spatiale avec le voisinage séquentiel. On les définit comme un couple de fragments proches en structure (et donc en interaction potentielle) correspondant au voisinage d'extension maximale le long de la séquence autour d'un contact.

Avant d'introduire une définition formelle, on peut penser intuitivement la définition de cette manière : au niveau d'un contact entre deux acides-aminés (*i.e.* distance entre les atomes C_α inférieure à un seuil σ), on fixe une ficelle de longueur τ (correspondant à la distance maximale pour avoir une potentielle interaction entre deux acides-aminés), une extrémité sur chacun des acides-aminés du contact. Le CF associé au point de départ de la ficelle est alors défini comme la zone du squelette où les extrémités de la ficelle peuvent circuler successivement d'un acide-aminé au suivant (voir figure 6.1). Afin de considérer seulement les contacts liés au repliement de la structure (*i.e.* acides-aminés séquentiellement distants), on impose de plus qu'il y ait au moins 4 acides-aminés le long de la séquence entre les deux extrémités de la ficelle.

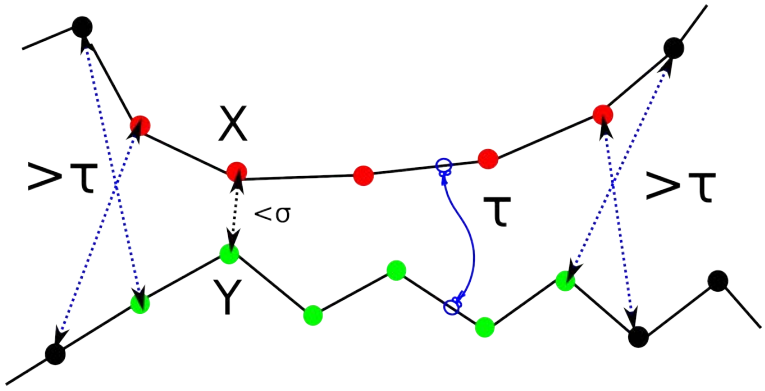

FIG. 6.1: Illustration intuitive de la définition d'un CF : la ficelle de longueur τ (en bleu) peut circuler d'acide-aminé en acide-aminé le long du squelette. Elle peut ainsi parcourir les acides-aminés colorés en vert et en rouge, mais est trop courte pour atteindre les acides-aminés en noir.

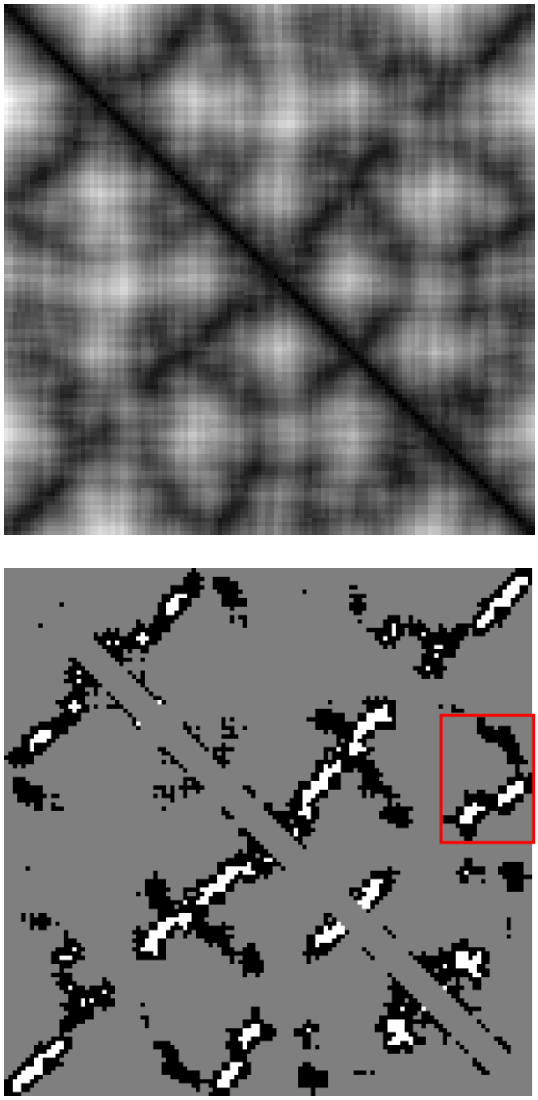

FIG. 6.2: Haut : matrice des distances internes de la protéine 2M5S. Plus un pixel est foncé, plus la distance $d_{i,j}$ est petite. Bas : Les pixels correspondant aux contacts sont en blanc, ceux correspondant à une distance inférieure à τ sont en noir. Un CF est défini par une composante connexe noire possédant une graine (*i.e.* un contact, en blanc). Un CF est par exemple encadré en rouge.

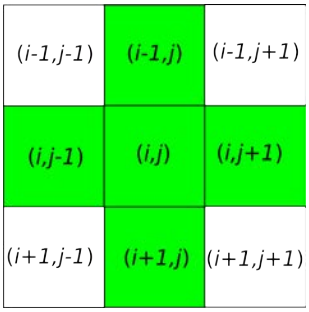

FIG. 6.3: Voisinage de von Neumann (en vert) de l'élément (i, j).

On peut visualiser plus formellement le déplacement de la ficelle grâce à la matrice des distances internes (figure 6.2 en haut) : de cette matrice, on supprime la partie autour de la diagonale correspondant à la contraintes des 4 acides-aminés entre les deux extrémités de la ficelle. Puis on établit deux coupures représentées en couleur blanche et noire en bas de la figure 6.2, correspondant respectivement aux seuils σ et τ. Chaque cellule blanche de coordonnées (i, j) correspond donc à un contact, point de départ de la ficelle : on peut mettre une extrémité sur l'acide-aminé i et l'autre sur l'acide-aminé j. La composante connexe noire entourant un point (i, j) définit les points correspondants aux déplacements possibles de la ficelle le long du squelette (en considérant que deux cellules de la matrice sont voisines si elles sont adjacentes et qu'on peut se déplacer de l'une à l'autre sur un axe vertical ou horizontal, aussi appelé voisinage de Von Neumann illustré en figure 6.3).

Une telle composante connexe est entourée par un rectangle rouge sur la figure 6.2. Ce rectangle définit par son bord vertical un intervalle $[\![x_m, x_M]\!]$ sur la séquence (resp. $[\![y_m, y_M]\!]$ par son bord horizontal), et le CF correspondant à cette composante connexe noire est la portion de structure définie par l'ensemble des deux fragments de structures issues des positions $[\![x_m, x_M]\!]$ et $[\![y_m, y_M]\!]$ (la structure de ces deux fragments est respectivement représentée en rouge et en vert sur la figure 6.4).

FIG. 6.4: Exemple de fragments en contact (CF) dans la protéine 2M5S. Le premier fragment suivant l'ordre de la séquence est en rouge et le second est en vert.

Voyons à présent quelques définitions pour introduire formellement les CF.

Par mesure de simplicité et pour ne pas reposer sur des outils de prédictions d'interactions, nous définissons un *contact* de la manière suivante :

Définition 6.1. *Deux acides-aminés a_i et a_j sont dits en* contact *si la distance entre leurs atomes C_α est inférieure à un certain seuil σ (généralement pris à 7.5Å, sauf mention contraire).*

Le seuil σ fixé à 7.5Å est commun dans la littérature traitant des contacts dans les structures de protéines car il correspond à un fort taux de probabilité d'avoir une interaction physico-chimique (pont hydrogène, pont disulfure) entre deux acides-aminés.

Nous rappelons ici que nous notons $p_1, ...p_N \in \mathbb{R}^3$ la structure d'une protéine de taille N et qu'on dénote D_P la matrice des distances internes de la protéine P ($D_{P_{i,j}} := d(p_i, p_j)$ où d est la distance euclidienne usuelle dans $\mathbb{R}^3$). Pour formaliser la définition de fragments en contacts, nous introduisons le graph G_P^τ d'une protéine P capturant les couples d'acides-aminés potentiellement en interaction :

Définition 6.2. *On note G_P^τ le* graph d'interaction *dont l'ensemble V des sommets sont les paires (i,j) telles que $|i-j| > 3$ et telles que $D_{P_{i,j}} \leq \tau$ et dont l'ensemble E des arrêtes est défini par les voisins immédiats selon l'ordre dans la séquence (voisinage dit de Von Neumann (voir figure 6.3)) :*

$$(i,j) \rightarrow (k,l) \in E \text{ si et seulement si } (i,j) \in V \wedge (k,l) \in V \wedge |i-k| = 1 - |j-l|$$

On notera $\mathcal{C}(i,j)$ la composante connexe complète dans G_P^τ contenant le sommet (i,j).

La contrainte $|i-j| > 3$ permet de s'affranchir des interactions entre acides-aminées qui ne sont pas liées au repliement de la structure mais uniquement à la proximité dans la séquence.

Définition 6.3 (Rectangles)**.** *Le* rectangle $R(x_m, x_M, y_m, y_M)$ *est l'ensemble* $\{(i,j) | x_m \leq i \leq x_M \wedge y_m \leq j \leq y_M\}$.

Définition 6.4 (Fragments en contact - CF)**.** *Une paire de fragments en contact (CF pour contact fragments) d'une protéine P est la portion de structure définie par le couple d'intervalles $([\![x_m, x_M]\!], [\![y_m, y_M]\!])$ (avec $x_m \leq y_m$) définis par le plus petit rectangle $R(x_m, x_M, y_m, y_M)$ contenant la composante connexe $\mathcal{C}(i_c, j_c)$ dans G_P^τ d'un contact (i_c, j_c). Le segment $[\![x_m, x_M]\!]$ est appelé segment gauche, et $[\![y_m, y_M]\!]$, segment droit.*

La proposition suivante justifie le nom de "fragments en contact", et montre que la définition donnée dans précédemment capture bien la notion intuitive de "deux segments en interaction potentielle" :

Proposition 6.1.1. *Pour tout CF défini par les résidus x_m, x_M et y_m, y_M :*

Contact $\exists i \in [\![x_m, x_M]\!], \exists j \in [\![y_m, y_M]\!]$ *tels que* $d_{i,j} \leq \sigma$

Interaction gauche-droite $\forall i \in [\![x_m, x_M]\!], \exists j \in [\![y_m, y_M]\!]$ *tel que* $d_{i,j} \leq \tau$

Interaction droite-gauche $\forall j \in [\![y_m, y_M]\!], \exists i \in [\![x_m, x_M]\!]$ *tel que* $d_{i,j} \leq \tau$

Démonstration. La présence d'un contact dans un CF est immédiate par la définition d'un CF. Pour montrer que le segment gauche est en interaction avec le segment droit, il faut remarquer qu'un CF est défini par le plus petit rectangle contenant une composante connexe de G^τ contenant un contact. Considérons un tel CF défini par un rectangle $R(x_m, x_M, y_m, y_M)$ à partir d'un contact (i_c, j_c). S'il existait un $x \in [\![x_m, x_M]\!]$ tel que $\forall j \in [\![y_m, y_M]\!], d_{i,j} > \tau$, alors aucune paire (x, j) n'appartiendrait aux sommets de G^τ (par définition de G^τ), et :

- ou bien $x_m < x < x_M$ et dans ce cas il ne peut avoir de chemin dans G^τ entre les points de $(i_<, j) \in G^\tau$ avec $i_< < x$ et les points $(i_>, j) \in G^\tau$ avec $i_> > x$. Alors $R(x+1, x_M, y_m, y_M)$ et $R(x_m, x-1, y_m, y_M)$ seraient deux rectangles strictement plus petit que $R(x_m, x_M, y_m, y_M)$ dont l'un des deux contiendrait $\mathcal{C}(i_c, j_c)$.
- ou bien $x = x_m$ (resp. $x = x_M$) et $R(x+1, x_M, y_m, y_M)$ (resp. $R(x_m, x-1, y_m, y_M)$) serait un rectangle strictement plus petit que $R(x_m, x_M, y_m, y_M)$ contenant $\mathcal{C}(i_c, j_c)$.

Dans les deux cas, on nierait le critère de minimalité du rectangle $R(x_m, x_M, y_m, y_M)$.
La démonstration pour le cas droite-gauche est similaire.

□

Il est par ailleurs intéressant de remarquer que les CF sont définis de manière intrinsèque à une structure. En effet, lorsque σ et τ sont fixés, à une structure est automatiquement associé son ensemble de CF. Par ailleurs, étant donné la définition des CF, il existe une correspondance fonctionnelle (au sens mathématique) entre les contacts et les CFs : à chaque contact est associé un CF. Par contre, cette relation n'est pas injective : un CF peut contenir plusieurs contacts.

Nous présentons maintenant quelques statistiques sur les caractéristiques (nombre, longueur, etc.) des CFs sur le jeu de données (Astral64, détaillé en annexe 9) représentatif des 4 principales classes SCOP.

6.2 Influence du paramètre τ

Les CF sont définis par deux seuils de distance σ et τ entre atomes de C_α. Le premier est le seuil définissant un contact, le second définit la zone d'interaction mutuelle des segments de structure. Le seuil σ est lié à des considérations physiques impliquant le rayon de Van der Waals des atomes de la chaîne latérale, alors que le seuil τ est davantage arbitraire et dépend du problème biologique posé. En effet, on peut vouloir fixer un seuil τ de l'ordre de 13Å afin de capturer par un seul CF deux hélices α dont les bords sont liés par des liaisons hydrogène, ou on peut vouloir un seuil plus fin de l'ordre de σ afin de focaliser une étude sur les parties de la structure en forte interaction. Ceci permet par exemple de pouvoir reconnaître des protéines partageant une similarité locale (*i.e.* ses CF seront similaires pour des petites valeurs de τ) mais différentes à une échelle globale (*i.e.* CF différents pour des plus grandes valeurs de τ).

Nous montrons dans un premier temps les distributions de taille et de nombre de fragments en contact que l'on peut obtenir suivant différentes valeurs de τ fixées. Nous verrons ensuite comment on peut extraire et organiser tous les fragments en contact d'une protéine sans fixer a priori la valeur τ.

6.2.1 Distributions avec τ fixé

Chevauchement On peut remarquer dans la définition que rien n'empêche les segments gauches et droits d'être chevauchants (*i.e.* le segment gauche finit après le début du segment droit). Dans Astral64 (sous-ensemble représentatif des 4 principales classes de SCOP, voir description du jeu de données en annexe 9), il y a par exemple 32% des CFs qui sont chevauchants. La figure 6.5 montre la distribution de la valeur du gap (*i.e.* $y_m - x_M$) ; on remarque le pic de CF ayant un gap de 4, ce qui correspond à la limitation $|i - j| > 3$ dans la définition de G_P^τ.

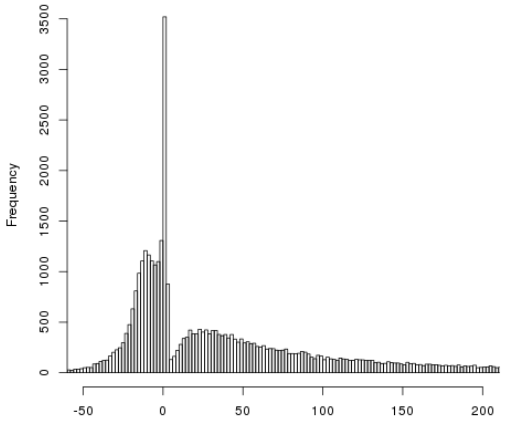

FIG. 6.5: Distribution du *gap* entre la fin du segment gauche et le début du segment droit ; les valeurs négatives correspondant aux CFs chevauchants (32% de la distribution).

Les CF chevauchants peuvent être considérés comme des cas dégénérés de CF : ils consistent en réalité en un seul fragment contigu. Pour certaines expériences où l'on veut tester les interactions séquentiellement distantes (voir section 6.4 par exemple) nous conserverons les CF constitués de deux segments non chevauchants.

Longueurs et quantité de fragments Les figures suivantes montrent la distribution — sur le jeu de données Astral64, voir détail en annexe 9 — de la longueur des segments gauches des CFs (A) ainsi que le nombre de CFs (B et C) par protéines en fonction de la valeur de τ et de la longueur des séquences :

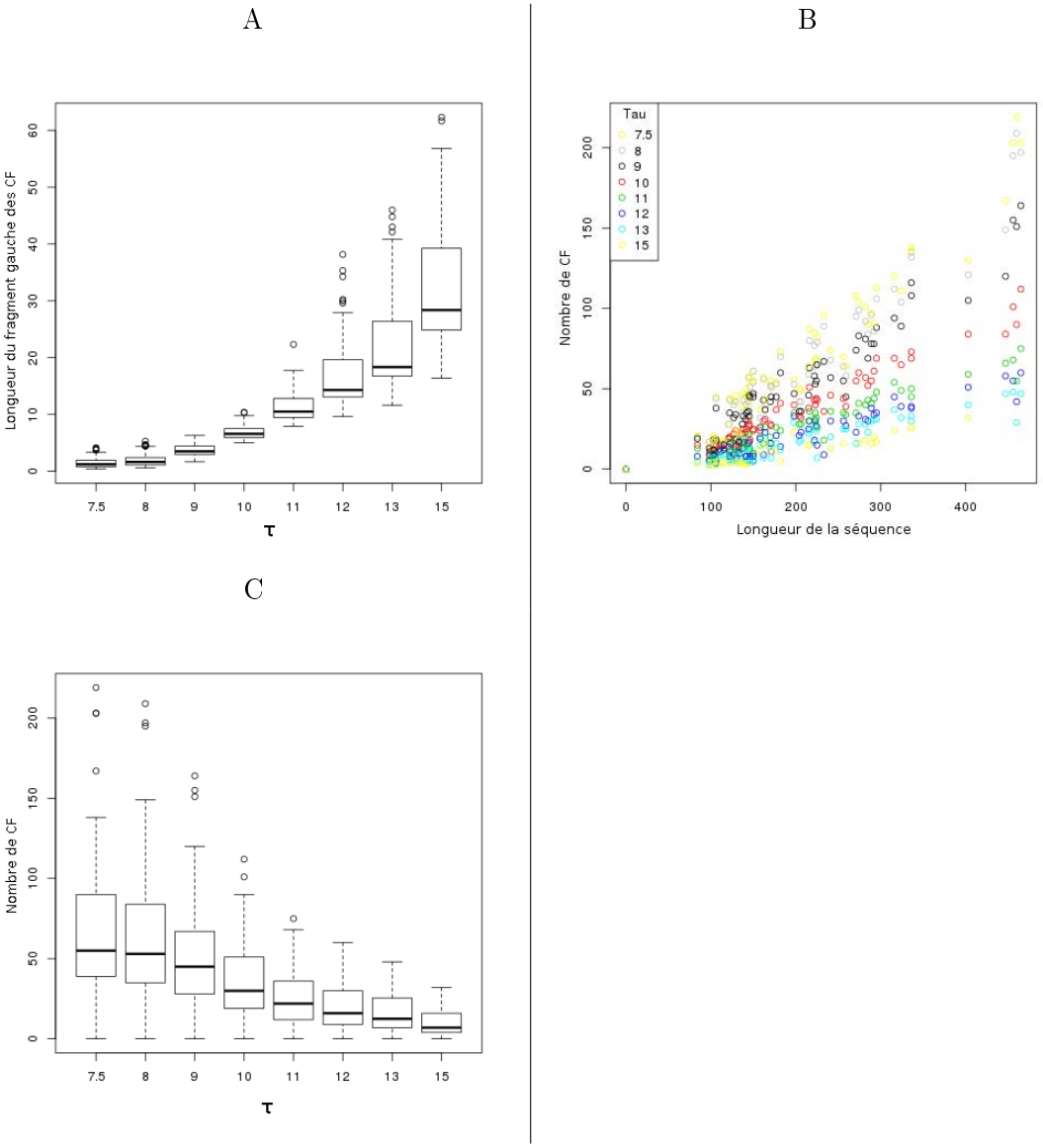

FIG. 6.6: Statistiques réalisées sur le jeu de données Astral64. A : longueur du segment gauche des CF ; B : nombre de CF en fonction de la longueur de la séquence et de la valeur de τ ; C : nombre de CF en fonction de τ

On voit qu'à τ fixé, le nombre de CF augmente linéairement avec la longueur de la séquence. Par contre, le nombre, mais aussi la variabilité du nombre de CF, augmente à mesure que τ diminue. Sauf mention contraire nous utiliserons dans les futures expérimentations la valeur standard de $\sigma = 7.5$Å utilisée dans les outils de génération de carte de contact, et la valeur de $\tau = 13$Å permettant de capturer l'interaction de deux hélices α liées par des liaisons hydrogène en un seul CF.

6.2.2 Représentation hiérarchique

Dans la définition d'un CF, plus τ est petit, plus les motifs sont petits et nombreux. En effet, à mesure que τ augmente, deux motifs peuvent être amenés à fusionner comme sur l'exemple figure 6.7.

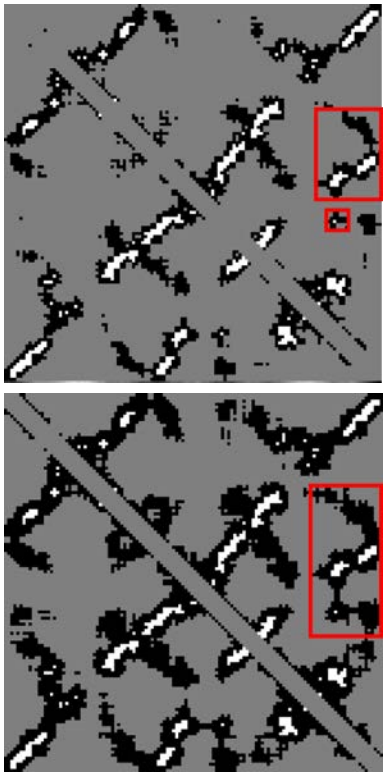

FIG. 6.7: Exemple de deux CF pour $\tau = 13$Å (haut) qui fusionnent à $\tau = 16$Å. Protéine 2M5S.

Afin d'illustrer concrètement l'organisation hiérarchique des fragments en contact, on montre ici un algorithme permettant d'extraire sous forme d'une arborescence binaire l'imbrication des CF d'une protéine à mesure que le τ augmente. Ainsi, on a une vision hiérarchique de l'assemblage de la structure d'une protéine qui peut être vu comme une extension des cartes de contact.

L'idée de l'algorithme est la suivante : un CF est défini par un contact, et deux contacts sont dans le même CF seulement s'il existe un chemin dans G^τ les reliant. Ainsi, le τ minimum correspondant au regroupement de deux contacts dans un même CF correspond à la valeur minimum du maximum $d_{i,j}$ rencontré le long de chaque chemin dans la matrice des distances internes entre les deux contacts. Ce problème, parfois dénommé *minimax path problem* en anglais, est le dual du problème classique de capacité maximum le long d'un chemin dans un graph.

Pour résoudre ce problème, on construit dans un premier temps un graph *non orienté*

G_d à partir de D, la matrice $N \times N$ des distances internes de la protéine P. G_d est un graph généralisant les graph G^τ pour tout τ : deux sommets dans G_τ seront reliés si et seulement si les deux sommets sont reliés dans G_d par une arrête dont le poids est inférieure à τ.

Pour éviter toute confusion, nous présentons l'algorithme de construction de G_d :
AjouterSommets($\{(i,j)|0 < i,j \leq N\}$)
Pour $0 < i < N$:
  Pour $0 < i < N$:
    AjouterArrête(entre (i,j) et $(i+1,j)$, poids= $\max(D_{i,j}, D_{i+1,j})$)
    AjouterArrête(entre (i,j) et $(i,j+1)$, poids= $\max(D_{i,j}, D_{i,j+1})$)
  Fin pour
Fin pour

L'algorithme de construction de l'arborescence des CF est alors :
$MST \leftarrow$ Arbre couvrant de poids minimal de G_d.
Pour chaque contact c_A :
  Pour chaque contact c_B :
    # c_A et c_B sont des couples (i_A, j_A) et (i_B, j_B) d'acides-aminés
    $p_{c_A,c_B} \leftarrow$ plus court chemin dans MST entre c_A et c_B
    $\tau_{c_A,c_B} \leftarrow$ valeur maximale du poids sur les arrêtes de MST le long de p_{c_A,c_B}
  Fin pour
Fin pour
 # La matrice τ définit une ultramétrique[1] *entre les contacts : $d(c_A, c_B) := \tau_{c_A,c_B}$. On peut donc la transformer en arbre.*
$T \leftarrow$ **ArbreDepuisUltramétrique**(τ)
Retourner(T)

On obtient alors une arborescence binaire, les feuilles représentant les contacts et la valeur des nœuds étant égale à la valeur de τ minimale pour rassembler les feuilles sous ce nœud dans un même CF. La complexité de création de l'arbre couvrant de poids minimum avec l'algorithme de Kruskal est de $O\big(N^2 \log N\big)$ avec N le nombre d'acides-aminés, on peut ensuite déterminer les distances entre les contacts en parcourant en temps linéaire pour chaque contact le MST ayant N^2 sommets, c'est-à-dire avec une complexité $O\big(C.N^2\big)$ où C est le nombre de contacts dans la structure. Enfin, la procédure **ArbreDepuisUltramétrique** utilise le tri des valeurs de la matrice τ, avec une complexité en $O\big(C^2 \log C\big)$, puis une création d'arbre par UnionFind également en $O\big(C^2 \log C\big)$. Au final, l'algorithme a une complexité de $O\big(N^2 \log N + C.N^2 + C^2 \log C\big)$.

[1] En effet, pour trois contacts quelconques c_A, c_B, c_C, τ respecte l'inégalité ultramétrique :

$$\begin{aligned} \tau_{c_A,c_C} &= \max_{\text{le long de } p_{c_A,c_C}} MST \\ &\leq \max\left[\max_{\text{le long de } p_{c_A,c_B}} MST, \max_{\text{le long de } p_{c_B,c_C}} MST\right] \\ &= \max(\tau_{c_A,c_B}, \tau_{c_B,c_C}) \end{aligned} \tag{6.1}$$

.

On peut borner l'expression précédente par la complexité $O(N^3)$.

FIG. 6.8: Exemple illustrant une partie de l'arborescence de CF extrait de la protéine 2M5S. Les nœuds sont nommés par le rectangle définissant le CF suivi du τ correspondant ; le dernier numéro étant un index identifiant de manière unique le CF dans la protéine.

6.3 Comparaison structurale de CF

Comme tout fragment de structure protéique, les CF peuvent être comparés avec des outils standards de comparaison de structure. Cependant, étant donné leur structure particulière sur deux fragments de squelette, nous verrons que de nouvelles mesures peuvent être plus adaptées.

Nous présenterons deux nouvelles méthodes de comparaison spécifiquement adaptées à la comparaison de CF. La première se base sur l'ASD introduit dans le chapitre 5, la seconde est une adaptation du programme Yakusa [CBP05] pour la recherche rapide de similarité de structure dans de grandes bases de données structurales (capable de fouiller la PDB non-redondante en un temps de l'ordre de la minute sur un ordinateur standard).

6.3.1 ASD pour les CF : ASD_{CF}

La définition des CF fait intervenir un seuil τ influençant la position des extrémités des CF. Dans deux structures similaires, il se peut que les CF analogues aient plus ou moins un acide-aminé au niveau de leur extrémité. Ainsi, le score doit être capable de comparer des structures de longueur différentes, et on ne peut utiliser un score qui comparerait les CF en présupposant un alignement un-à-un des acides-aminés des CF (le premier acide-aminé d'un CF pouvant en réalité correspondre au second, voire au

troisième d'un CF analogue dans une autre protéine). Ainsi, on peut penser qu'il serait judicieux de réaliser une phase d'alignement des CF avant de pouvoir les comparer. Nous avons vu dans le chapitre précédent que ceci était problématique d'une part s'il existe une distorsion dans les structures empêchant la superposition de celles-ci mais aussi pour des questions de qualité de classification (car on compare alors uniquement des sous-parties de structure, niant nécessairement la propriété d'inégalité triangulaire).

C'est pourquoi on se propose de définir l'ASD_{CF} permettant la comparaison de CF dans la même idée que l'ASD. Une implémentation a été réalisée en Python en utilisant la librairie standard *scipy* pour la transformée de Fourier.

Nous avons vu que la définition des CF fait intervenir des rectangles dans la matrice des distances internes. L'idée est d'utiliser uniquement la portion de matrice définie par ces rectangles pour la comparaison des CF. On compare avec l'ASD_{CF} les distances inter-fragments plutôt que les distances intra-fragment comme il a été présenté dans le cas de fragments simples du chapitre 5.

Formellement, on remplace simplement la matrice des distances internes par la matrices des distances mutuelles :

$$ASD_{CF}(CF_A, CF_B) := || \, |\mathcal{F}\widetilde{D_A^M}| - |\mathcal{F}\widetilde{D_B^M}| \, ||_2 \tag{6.2}$$

où $D^M_{X\,i,j} := d(l_i, r_j)$ et $\widetilde{D_X^M}$ sa version 0-complétée, l_i et r_j désignent respectivement sur le CF X, l'atome C_α de l'acide-aminé en position i sur le segment gauche et en position j sur le segment droit. d est la distance Euclidienne usuelle dans $\mathbb{R}^3$.

Par abus de notation, s'il n'y a pas possibilité de confusion, dès lors qu'on comparera des CF en utilisant l'ASD_{CF}, on notera ASD(CF_A, CF_B) au lieu de $ASD_{CF}(CF_A, CF_B)$.

Nous utiliserons cette mesure de similarité notamment pour des tâches de classification, en section 7.1 par exemple.

6.3.2 Adaptation de Yakusa pour les CF

Yakusa [CBP05] est un outil efficace pour la fouille de structure à grande échelle. Chaque fragment de 4 acides-aminés consécutif est codé ici par la discrétisation de l'angle interne α [Lev76] formé par les atomes C_α le long du squelette. Une structure est alors représentée par la séquence de ces symboles associés à chaque fragment chevauchant de longueur 4.

Le principe de comparaison repose ensuite sur le même que celui de BLAST (voir section 2.1.0.6) pour la séquence : *i)* identification rapide de graines structurales identiques (*hit*) à celles contenues dans la requête dans la base à chercher, *ii)* extension des zones identifiées le long du squelette, *iii)* calcul d'un score de similarité associé à chaque *hit*.

La modification de Yakusa pour la comparaison de CF a été réalisée dans le cadre du stage de Master 2 [Jus15]. L'idée est de chercher indépendamment chaque segment avec la méthode standard de Yakusa puis de vérifier la compatibilité de leur distance mutuelle *a posteriori*. Plus précisément, on vérifie que les distances respectives entre les acides-aminés des extrémités des segments ainsi que de l'acide-aminé médian de chaque segment sont similaires entre la requête et chaque structure identifiée dans la base.

Cette adaptation permet d'identifier dans la PDB non-redondante les structures similaires à un CF en un temps de l'ordre de la minute. Ce mode de comparaison ne permet pas d'identifier finement, comme avec l'ASD, des structures globalement similaires qui auraient des distorsions locales, mais permet néanmoins de fouiller efficacement la PDB entière. Nous l'utiliserons notamment lorsque nous chercherons à identifier des CF spécifiques d'une famille (cf. section 7.1).

6.4 Prédictibilité des CF

Cette section étudie l'influence de la localité spatiale sur la prédictibilité d'une structure. Est-ce qu'une mutation dans une zone de séquence en contact est souvent destructrice de la structure ? Est-ce qu'au contraire il existe une redondance dans les interactions autorisant un taux de mutation élevé sans changement de structure ? Est-ce qu'il existe dans les protéines plus de diversité de séquence pour une même structure lorsque celle-ci est en interaction avec une autre partie de la structure ?

L'expérience réalisée ici pour répondre à ces questions consiste à comparer la prédictibilité des CF à celle des des *fragments simples* (notés SF pour *Single Fragments*) ainsi qu'à des *paires de fragments* qui ne seraient pas en interaction (notés PF pour *Pair of Fragments*). A cette fin, nous avons extrait tous les CF du jeu de données Astral64 (voir détail de ce jeu de données en annexe 9), ainsi que pour chaque CF une paire de fragments (PF) possédant les même longueurs de segments, mais pris aléatoirement dans la séquence. Aussi, nous nous sommes limités aux CF n'ayant pas de chevauchement entre les segments et nous avons choisi les PF de manière à ce qu'il n'y ait là encore aucun chevauchement entre les segments. Enfin, pour chaque CF non chevauchant, nous avons extrait un fragment simple (*i.e.* contigu en séquence) pris aléatoirement dans la séquence dont la longueur vaut la somme des longueurs des deux fragments de CF.

Ensuite, pour pouvoir comparer la conservation de structure en fonction de la conservation de séquence entre les CF, les PF et les SF, l'idée est pour chaque CF (resp. PF, SF) de chercher dans la PDB — avec BLAST — les structures ayant une *séquence* similaire. On mesure ensuite la similarité de *structure* entre le CF (resp. PF, SF) requête et le *hit* dans la PDB. Ce procédé peut se résumer sous le diagramme suivant :

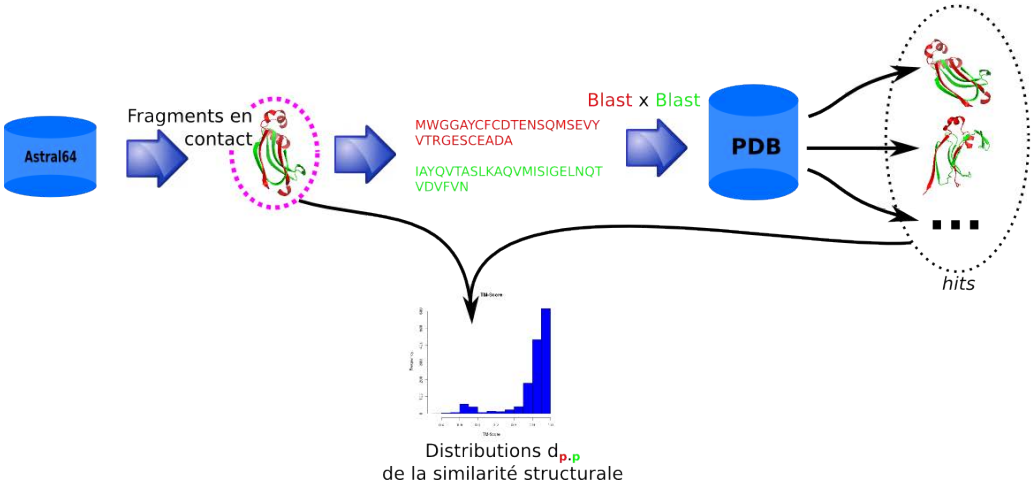

FIG. 6.9: Procédure utilisée pour la mesure de la prédictibilité faible des CF. La procédure *identique* est réalisée avec les PF (paires de fragments) et les SF (simples fragments).

Plus précisément, pour les CF et PF, chaque segment est cherché indépendamment avec BLAST et une p-value est calculée — pour chaque segment — à partir de la e-value de BLAST. La p-value totale est définie comme étant la multiplication des p-values de chacun des segments (car ceux-ci sont non chevauchant, leur *hits* sont considérés indépendants). Pour les SF, il existe un biais lié à l'heuristique de BLAST : la recherche de séquence de SF nécessite une seule graine (voir section 2.1.0.6) alors que la recherche des deux segments des CF et des PF nécessite deux graines (une pour chaque segment). En conséquence, il y aura plus de *hits* pour les SF que pour les CF et les PF et en moyenne dans un hit l'identité de séquence sera probablement biaisée vers des valeurs légèrement plus élevée dans les CF et PF.

Par ailleurs, étant donné qu'on cherche à comparer structurellement un CF (resp. PF, SF) requête avec son hit, les fragments à comparer sont de même longueur et on peut utiliser les scores standards de similarité de structure (nous montrons les résultats avec le TM-score qui a l'avantage d'être normalisé). Utiliser un outil standard de comparaison de structure permet également d'éviter d'introduire biais dans l'analyse.

On obtient donc pour chaque *hit* un couple de valeurs (p-value,TM-score). En prenant tous ces couples pour lesquels la p-value est inférieure à un certain seuil p_{max}, on obtient pour chaque p_{max} la distribution $TM_{CF}(p_{max})$ (resp. $TM_{PF}(p_{max})$, $TM_{SF}(p_{max})$) des valeurs du TM-score pour les CF (resp. pour les PF, les SF).

Afin d'étudier ces multiples distributions, on a choisi de regarder les différences dans le haut des distributions associées aux CF et celles associées aux PF — les bas et médianes de distributions étant semblables. Les résultats sont visibles sur la figure 6.10.

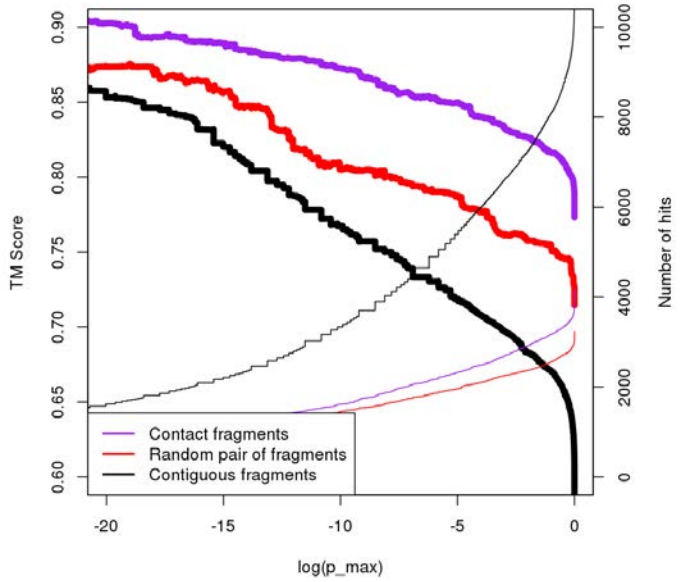

FIG. 6.10: En gras : Dernier 5-quantile (80% des valeurs sont inférieures) de $TM_{CF}(p_{max})$ (en violet), $TM_{PF}(p_{max})$ (en rouge), $TM_{SF}(p_{max})$ (en noir) en fonction de p_{max}. En ligne minces : nombre de *hits* ayant une p-value inférieure à p_{max} en fonction de p_{max}.

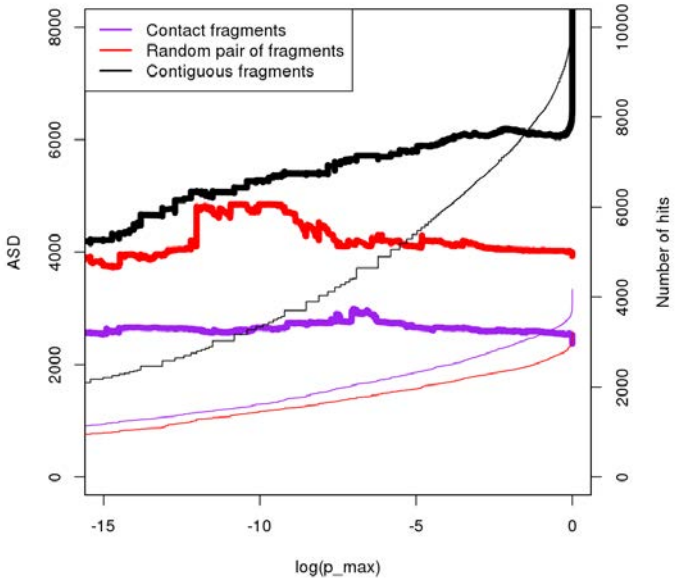

FIG. 6.11: Même graphique que précédemment en utilisant l'ASD à la place du TM-Score. Les conclusions sont comparables, l'ordre des courbes étant inversé car l'ASD est une dissimilarité alors que le TM-Score est une similarité de structure.

Sur le graphique 6.10, on peut voir que si on désire trouver l'ensemble d'homologues (en séquence) d'un motif structural tel que sa structure soit correctement conservée, alors il faudra fixer un seuil de séquence beaucoup plus strict si le motif est un SF ou un PF que si c'est un CF. Par exemple, s'il on veut que 80% des *hits* aient une structure fortement conservée (par exemple un TM-score > 0.85), il faudra fixer un seuil de similarité de séquence correspondant à une p-value $p_{max} \approx 10^{-18}$ pour les SF, de $p_{max} \approx 10^{-14}$ pour les PF, et de $p_{max} \approx 10^{-5}$ pour les CF. On voit sur ce même graphique, qu'avec ces seuils, on obtiendrait environ 1800 hits de SF, 1000 hits de PF et environ 2500 hits de CF.

On peut conclure que sur ce sous-ensemble de structure représentatif des 4 principales classes SCOP, pour une même structure (à faible distorsion près), la diversité de séquences possibles dans le cas des CF est plus importante que dans le cas des SF et des PF.

Pour reprendre les termes de la formalisation du chapitre 4, nous avons évalué ici la *cohérence* de la prédictibilité des CF avec pour mesure de dissimilarité de structure $d(A, B) := 1 - TM(A, B)$, et pour modèle $M_{t_{CF}} \vdash q \Leftrightarrow p\text{-}value(q, q_{CF}) \leq p_{max}$ où $p\text{-}value(q, r)$ est la p-value associée à la similarité de séquence entre q et r et t_{CF} est une structure de CF de séquence q_{CF} extraite d'une structure d'Astral64. Ainsi, 80% des structures homologues (pour un seuil de p-value $p_{max} = 10^{-5}$) ont un TM-Score avec la structure requête supérieur à 0.85, ce qui peut se reformuler en disant que la règle de

cohérence est respectée à 80% par les CF pour des paramètres $\delta \approx 1 - 0.85 = 0.15$ et $p_{max} = 10^{-5}$, alors qu'elle sera respectée à 80% par les SF pour $\delta \approx 1 - 0.72 = 0.28$, et par les PF pour $\delta \approx 1 - 0.79 = 0.21$.

L'expérience exposée montre donc que pour un CF on retrouve plus de diversité de séquence que dans un fragment simple ou dans une structure composée d'une paire de segments qui ne sont pas spécifiquement en interaction. Il semble donc qu'il existe une certaine redondance dans les interactions permettant le maintien de la structure et que la mutation de la séquence sous-jacente tolère davantage de mutations sans changement de conformation. Selon les termes du chapitre 4, la localité spatiale renforce la cohérence de la prédictibilité (pour un modèle de séquence basé sur l'homologie). Pour aller plus loin, on pourrait utiliser un modèle de séquence plus fin prenant en compte les mutations compensatoires dans les CF qui pourrait éventuellement améliorer encore leur prédictibilité.

6.5 Perspective : détection de CF par modèle logique de co-évolution

Nous avons commencé à explorer une approche logique pour la modélisation de séquences de CF. Étant donné que les CF se focalisent sur des portions de structures localisées dans l'espace, il doit exister une information dans leur séquence liée aux mutations compensatoires permettant le maintien de la structure au cours de l'évolution. Nous avons voulu exploiter ce fait en réalisant un modèle logique des séquences capturant les interactions entre acides-aminés dans les CF.

Le modèle logique proposé est simple : il s'agit de définir un prédicat $P(a, i)$ indiquant que l'acide-aminé a se situe en position i sur la séquence, et à partir d'un alignement multiple, de recenser toutes les implications valides de la forme $P(a, i) \Rightarrow P(b, j)$ étant au moins supportées par une séquence de l'alignement.

Un outil développé au cours de cette thèse en ASP (*Answer Set Programming* [GKK+11]), Python et Dot (pour la génération de graphs `www.graphviz.org`) permet de représenter graphiquement toutes les implications de la forme $P(a, i) \Rightarrow P(b, j)$.

Si on considère l'exemple jouet d'alignement multiple présenté en figure 6.12, notre outil calcul le modèle logique illustré en figure 6.13.

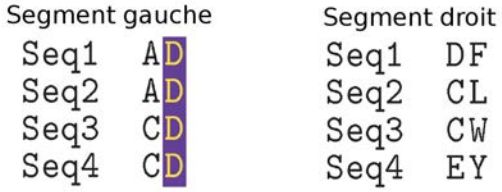

FIG. 6.12: Exemple jouet de séquences de CF.

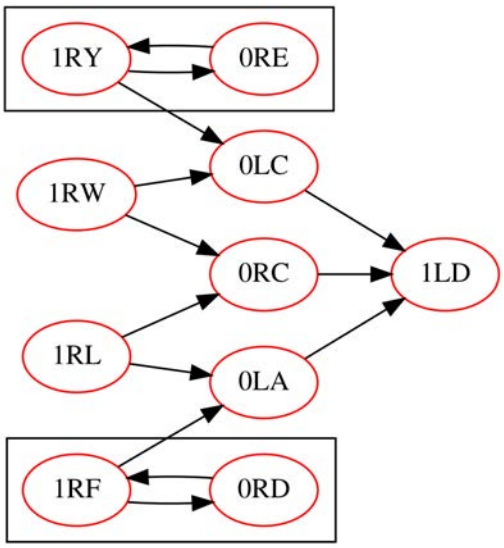

FIG. 6.13: Représentation logique des co-évolutions dans l'alignement multiple ci-dessus figure 6.12. Chaque noeud est un prédicat. Les étiquettes sont de la forme iLA (resp. iRA), signifiant que l'acide-aminé en position i sur le segment de gauche (resp. de droite) est A. Chaque flèche représente une implication, les blocs représentent les prédicats équivalents.

Une séquence sera validée par le modèle logique si toutes les implications sont respectées. C'est-à-dire que si la séquence respecte le prédicat $P(a, i)$ et que le modèle possède une implication $P(a, i) \Rightarrow P(b, j)$, alors la séquence doit respecter le prédicat $P(b, j)$. Il serait aussi possible d'établir un score de similarité de séquence correspondant au nombre d'implications respectées.

De manière à obtenir des modèles pouvant généraliser davantage l'ensemble de séquences reconnues, nous pouvons remplacer les acides-aminés par certaines de leurs propriété physico-chimiques. La figure 6.14 illustre un tel modèle basé sur l'exemple jouet de l'alignement multiple en figure 6.12. Ce modèle est similaire, mais plus général que celui exposé précédemment en figure 6.13.

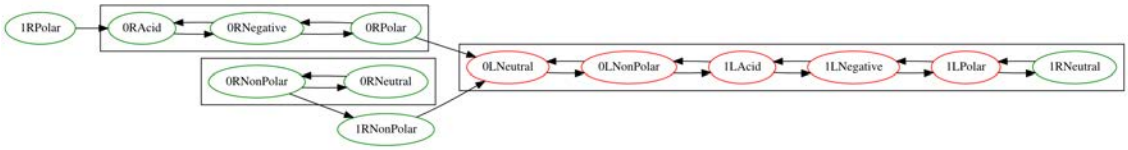

FIG. 6.14: Représentation logique des co-évolution dans l'alignement multiple ci-dessus en figure 6.12, en utilisant les propriétés physico-chimiques au lieu des acides-aminés.

Les conservations usuelles des alignements multiples se retrouvent en regardant les

```
Left2HWD_19   FGENMYYHFLGRSGYTVHVQCNASKFHQGTLLVAMIPEH
Left1AYM_74   FGENMFYHFLGRSGYTVHVQCNASKFHQGTLLVVMIPEH
Left3VDD_59   FGENMFYHYLGRSGYTIHVQCNASKFHQGTLIVALIPEH
Left1RHI_76   FGQNMFYHSLGRTGYTIHVQCNATKFHSGCLLVVVIPEH
Left1K5M_48   FGQNMFFHSLGRSGYTVHVQCNATKFHSGCLLVVVIPEH


Right2HWD_19  LGNLLIFPHQFINLRSNNSATLIVPYVNAVPMDSMLR.HNNWCLV
Right1AYM_74  LGNLLIFPHQFINLRSNNSATLIVPYVNAVPMDSMVR.HNNWSLV
Right3VDD_59  LGNITIFPHQFINLRSNNSATIIAPYVNAVPMDSM.RSHNNWSLV
Right1RHI_76  LGNLLIFPHQFINMRTNNTATIVVPYINSVPIDSMTR.HNNVSLM
Right1K5M_48  LGNLLIFPHQFINLRTNNTATIVIPYINSVPIDSMTR.HNNVSLM


Right2HWD_19  IIP
Right1AYM_74  II.
Right3VDD_59  IIP
Right1RHI_76  VVP
Right1K5M_48  VIP
```

FIG. 6.15: Exemple de séquences d'un groupe de CF extrait du jeu de capsides de virus. Le premier alignement représente l'alignement des séquences du segment gauche, et le second du segment droit.

éléments maximaux pour l'ordre partiel de l'implication (*i.e.* $a \leq b$ si et seulement si $a \Rightarrow b$). En figure 6.13, le nœud "1LD" indique la conservation de de l'acide-aminé D en colonne 1 dans le segment de gauche du CF.

On voit déjà que les relations peuvent être complexes pour un exemple aussi simple que celui de l'alignement en figure 6.12. Nous montrons en figure 6.15un exemple concret d'un alignement multiple d'un cluster de CF extrait du jeu de données de capsides et en figure 6.16 le modèle logique associé.

Il serait intéressant de comparer la cohérence de telles représentations logiques de séquence avec la cohérence des modèles à base de HMM ainsi que ceux basé sur la simple similarité de séquence (comme dans l'expérience en section 6.4).

6.6 Conclusion

Nous avons introduit les fragments en contact comme des portions de structures qui concilient la localité spatiale avec le voisinage séquentiel. Nous avons montré sur un jeu de données représentatif des 4 principales classes SCOP que la prédictibilité des fragments en contact étaient plus élevée que celle des fragments simples ainsi que des paires de fragments qui ne seraient pas en contact en structure pour un modèle de séquence basé sur la similarité de séquence. On pourrait éventuellement affiner cette détection depuis la séquence en prenant en compte les éventuelles co-évolutions d'acides-

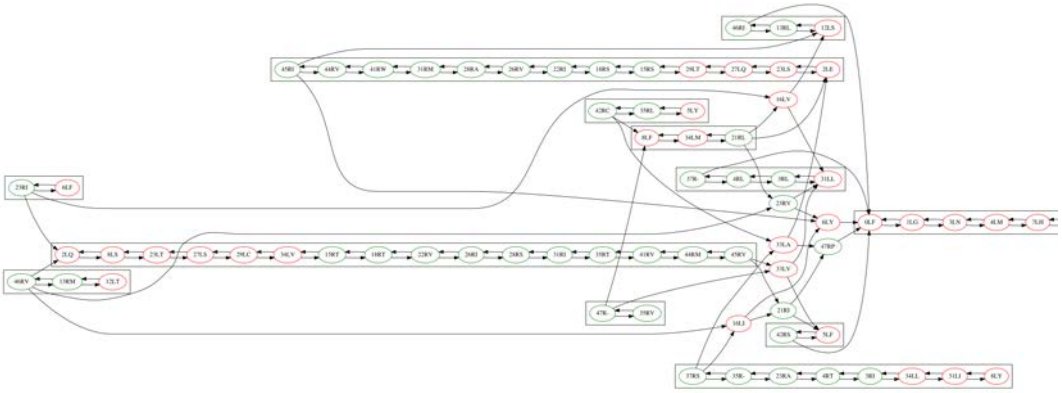

FIG. 6.16: Extrait de la représentation logique inférée à partir de l'alignement multiple (en figure 6.12) de séquences de CF de capsides de virus.

aminés en interaction dans la structure des CFs. Une possibilité d'investigation a été présentée en section 6.5, dont la difficulté principale de mise en œuvre réside dans l'association d'un score au modèle logique de co-évolution proposé.

Dans le prochain chapitre nous présentons quelques applications de l'utilisation des CF sur des données réelles pour la caractérisation de protéines virales au niveau de la structure. Nous montrerons comment l'apprentissage automatique permet de tirer parti de la détection de CF pour l'identification en séquence de protéines structurales de virus.

Chapitre 7

Applications à la caractérisation de protéines structurales de virus

Les virus sont des entités biologiques de plus en plus reconnues pour jouer un rôle majeur dans l'évolution. Ils semblent en effet avoir largement permis des transferts horizontaux de gènes pouvant conférer des avantages sélectifs à des espèces sans que celles-ci ne les aient hérités d'un ancêtre ni même réinventés par homoplasie [Hol11]. Par ailleurs, les virus sont impliqués dans des phénomènes physiques à très grande échelle. On estime par exemple que dans les océans, 20% de la biomasse est tuée quotidiennement par des virus. Les organismes morts à l'issue d'une infection virale tombent alors par gravité dans le fond des océans, emportant avec eux autant de carbone qu'ils avaient consommé de CO_2 pour se constituer. Ce phénomène permet de décharger la surface des océans de son carbone et alimenter la pompe biologique à CO_2. Ainsi les virus contribuent indirectement à l'absorption du CO_2 atmosphérique [LKS$^+$14].

La particule virale entière pouvant infecter un hôte est appelée *le virion*. La plupart des virions (voire la totalité suivant la définition du mot virus) ont leur matériel génétique enveloppé par des protéines formant une *capside*. Ces enveloppes peuvent se présenter sous diverses formes géométriques, comme par exemple l'icosaèdre ou l'hélice. Les contraintes d'assemblage des protéines pour réaliser ces formes géométriques se traduisent en contraintes structurales au niveau de chaque protéine. Les protéines de capside sont donc supposées avoir une structure bien conservée. Ces structures peuvent néanmoins être bien différentes d'un virus à l'autre, mais aussi au sein d'un même virus : plusieurs types de protéines de structures peuvent être nécessaires pour former la capside du virus (comme par exemple des protéines de fibre formant des "pics" sur la capside, ou encore les différentes protéines VPx présentées en figure 7.1).

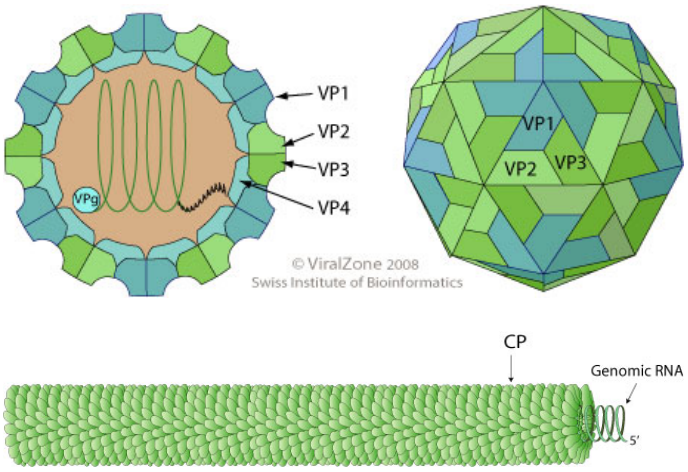

FIG. 7.1: Haut : capside icosaédrique d'*enterovirus*, les VPx sont les différentes protéines composant la capside. Bas : capside hélicoïdale de *tobamovirus* composée de l'assemblage d'un seul type de protéine (CP pour *capsid protein*). Source : ViralZone `viralzone.expasy.org`.

Malgré l'importance des virus, il n'existe pas de caractérisation structurale des protéines de capsides ; seuls quelques repliements ont été identifiés (comme le double *jelly roll* [KB11]), mais se retrouvent aussi bien chez d'autres protéines (comme par exemple chez les Glycosides Hydrolases). Nous présenterons dans la section 7.1 l'identification de CF spécifiques de structures de protéine de capsides. Ceux-ci peuvent alors constituer une base à la création de signatures structurales de capsides.

Au niveau de la séquence, les virus évoluent très rapidement [Dra99] et il n'est pas suffisant de se baser sur l'homologie pour détecter les séquences virales dans un génome ou métagénome [ER05]. Il est par exemple courant que des études métagénomiques sur les virus rapportent des taux d'annotation de seulement 10% [HS13, Sut07, HSS+13].

Cette problématique a été introduite dans cette thèse par la participation au projet PEPS VAG (en collaboration avec l'Atelier de Bioinformatique à l'Univeristé Pierre et Marie Curie et l'Insitut de Biologie de l'Ecole Normale Supérieure) dont l'objectif était l'annotation des séquences virales issues de l'expédition TaraOceans [PNP+15]. Une étude préliminaire a donné lieu à une présentation au *Workshop on Recent Computational Advances in Metagenomics* ECCB'14 [BCC+14].

L'organisation génomique et la séquence des gènes codant pour les protéines structurales (*i.e.* les protéines qui composent le virion, en particulier la capside et chez certains la queue) sont des critères pour la classification des virus. Nous présenterons en section 7.2 une méthode pour la détection de *séquences* de CF extraits de protéines de capsides. Ensuite, nous montrerons en section 7.3 l'utilisation faite de ces détecteurs dans l'outil VIRALpro [GMCB15] — développé au cours de cette thèse — permettant l'identification de séquences de protéines virales de capside et de queue.

7.1 Identification structurale de capsides à l'aide des CF

Nous nous intéressons ici à identifier des CF qui pourraient être spécifiques des structures de capsides. Lorsqu'une structure de CF est partagée par différentes structures de capsides et n'est retrouvé seulement dans des structures de capside, on dit que celui-ci constitue est un motif structural caractéristique. De tels motifs peuvent alors permettre l'annotation automatique de structures voire de séquences s'ils sont prédictibles.

Afin d'identifier les CF spécifiques des capsides, nous avons d'abord extrait l'ensemble des CF d'un ensemble de structures de capsides (jeu de données détaillé en annexe 9). Puis nous avons fouillé la PDB non redondante grâce à l'outil Yakusa (voir section 6.3.2) afin d'identifier les CF qui étaient seulement retrouvés chez les virus. Ce travail a été effectué en collaboration avec Maud Jusot [Jus15] dans le cadre du projet PEPS VAG.

Pour chaque CF, un ensemble de structures similaires est trouvé dans la PDB. La figure 7.2 montre deux composantes connexes du PowerGraph [RRAS08] de ces relations de similarité : une arrête connecte un CF ou un groupe de CF à un ou plusieurs identifiants PDB si ceux-ci possèdent une sous-structure similaire au CF. Dans un souci de visibilité, nous avons exposé uniquement les CF les plus caractéristiques, ou plus précisément tels que $\frac{Cap}{Cap + NonCap} > 90\%$ où Cap est le nombre de structures de capsides ayant une sous structure similaire au CF, et NonCap le nombre de non-capsides ayant une sous-structure similaire au CF. Sur la figure 7.2, les sommets verts correspondent à des structures de capsides, les sommets rouges aux structures n'étant pas des protéines de capside et les sommets gris sont des CF issus de structures de capsides.

Les sommets gris qui ne sont liés qu'à des sommets verts sont des CF spécifiques de capsides de virus. Ceux qui sont liés à des sommets rouges peuvent donc détecter d'autres types de protéines, mais pourraient éventuellement être spécifiques en combinaison avec d'autres CF (eux aussi éventuellement non-spécifiques).

Cette étude est restée à un stade préliminaire, néanmoins il est intéressant de remarquer que le graph présenté précédemment se décompose en composantes connexes spécifiques d'un type de virus voire d'un type de protéine de capside. Nous n'avons cependant pas identifié de CF spécifique à l'ensemble de toutes les protéines de capside : les structures peuvent être très différentes d'un type de virus à l'autre et d'un type de protéine à l'autre.

7.2 Détection des CF à partir de la séquence

Selon le vocabulaire du chapitre 4, détecter un CF depuis la séquence consiste à construire pour chaque structure t de CF un modèle M_t de l'ensemble séquences t'. Nous dirons que M_t détecte le CF t sur la séquence q si $M_t \vdash q$.

Nous présentons ici une méthode permettant une génération approchée de tels modèles à base de HMMs : à un ensemble de structures T qu'on considérera similaires, on associe un modèle de séquence M_T généralisant avec des HMMs les séquences des structures de T. Ces modèles seront construits à partir du jeu de données de capsides

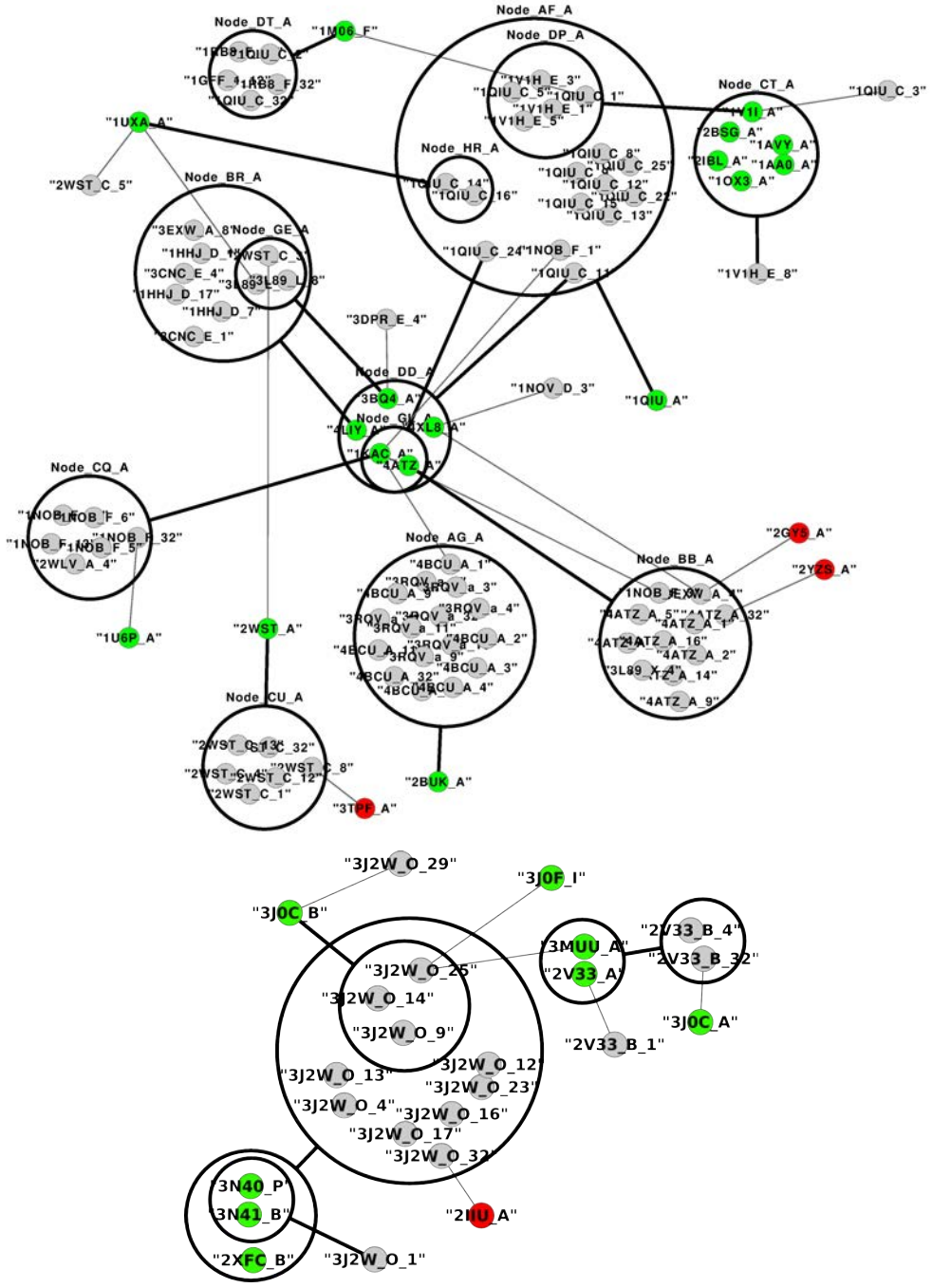

FIG. 7.2: Composantes connexes du PowerGraph formé par les CF caractéristiques des protéines de fibre sur les capsides (composante du haut : capsides d'Adenovirus, composante du bas : capsides d'Alphavirus). Les sommets gris qui ne sont liés qu'à des sommets verts sont des CF spécifiques de capsides de virus. Ceux qui sont liés à des sommets rouges peuvent donc détecter d'autres types de protéines, mais pourraient éventuellement être spécifiques en combinaison avec d'autres CF (eux aussi éventuellement non-spécifiques).

de virus (voir détails du jeu de données en annexe 9).

Afin de construire des modèles pour la détection de CF en séquence, nous cherchons dans un premier temps à regrouper les structures de CF similaires avec l'ASD. Plutôt que d'imposer un seuil arbitraire de similarité de structure (dénoté δ dans le chapitre 4), nous avons préféré classifier les CF de manière non supervisée. Pour obtenir *in fine* des groupes de CF similaires, nous avons pris dans la hiérarchie de la classification, en partant des feuilles, les clusters les plus larges possibles qui contenaient des CFs partageant la même taxonomie. Cette méthode ne garantit pas d'avoir au sein d'un cluster des CF parfaitement similaires, mais au contraire permet d'avoir des clusters de CF les plus généraux possibles tout en étant spécifiques d'une taxonomie. Dans notre cas, la taxonomie des virus est divisée suivant les grands groupes suivants :

- Virus à ADN double brin
- Virus à ADN simple brin
- Virus à ARN double brin
- Virus à ARN simple brin positif
- Virus à ARN simple brin négatif
- Rétro-virus

Nous avons ainsi extrait l'ensemble des CF (8423) du jeu de capsides — détaillé en annexe 9 — que nous avons regroupé avec une classification hiérarchique par une méthode de type Ward utilisant l'ASD comme distance sous-jacente entre les CF. En coupant le dendrogramme de classification selon la procédure que nous venons de décrire, nous avons obtenu 1420 clusters ayant entre 2 et 17 CF.

Pour chaque cluster nous pouvons alors extraire les séquences de ses CF et générer un modèle de séquence. Une approche directe serait de construire des HMMs à partir de l'alignement multiple des séquences de chaque cluster (voir figure 7.2). Cependant, comme nous l'avons fait remarquer plus haut, la diversité des séquences chez les virus est telle qu'un alignement entre deux séquences — bien que codant pour une structure similaire — peut être de très mauvaise qualité (il n'y a pas nécessairement d'homologie). Afin d'avoir des modèles de séquence plus fins, nous avons généré des alignements multiples de meilleure qualité.

Sous-alignements de meilleure qualité L'idée pour produire des alignements de meilleure qualité est de les réaliser sur des ensemble de séquences homogènes. Une idée simple serait de regrouper les séquences similaires entre elles et de faire plusieurs alignements multiples sur ces groupes. Le regroupement en séquences similaires imposerait cependant l'introduction de seuils de similarité, difficiles à évaluer automatiquement sur des séquences de conservation et de longueur variables.

Néanmoins, on peut remarquer le fait suivant : si deux groupes de séquences au sein d'un même cluster structural ne partagent aucune homologie, alors en regroupant les segments gauches d'une part (via une clusterisation hiérarchique par exemple), et les segments droits d'autre part, on obtiendrait à un certain niveau de coupure des deux arbres la situation présentée en figure 7.4. On peut ainsi déterminer des clusters "homogènes" : c'est-à-dire un cluster de fragments gauches regroupant les mêmes séquences qu'un cluster de fragments droits.

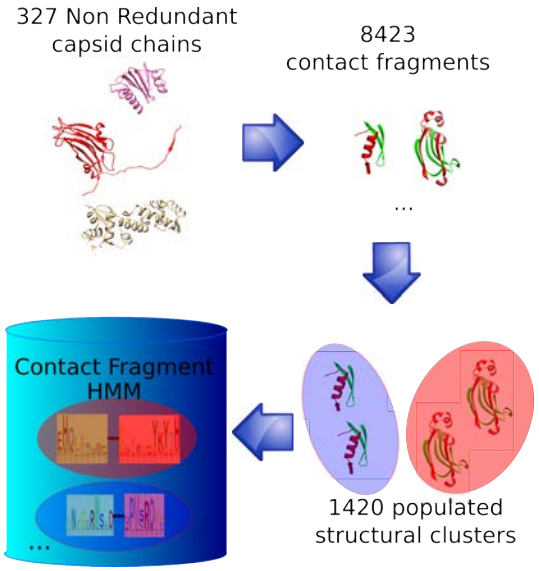

FIG. 7.3: Procédure standard pour la génération de HMM à partir de clusters de CF similaires. Ces HMM peuvent être considérés comme des prédicteurs de CF au niveau de la séquence.

FIG. 7.4: Classification de 5 séquences de CF obtenue en regroupant d'une part les fragments gauches et d'autre part les fragments droits par similarité de séquence, dans le cas de deux groupes différents de séquences homologues.

A un autre niveau de coupure (plus bas dans l'arbre), on pourrait obtenir la situation exposée en figure 7.5. Les clusters homogènes sont seulement les clusters des deux séquences du bas de la figure, contenant une seule séquence chacun.

FIG. 7.5: Clusters obtenus en coupant à un niveau plus fin dans l'arbre de classification hiérarchique.

Formellement, en classifiant hiérarchiquement les séquences des segments gauches (resp. droits), on obtient un ensemble de clusters C_i^L (resp. C_i^R). On dira qu'un ensemble X de CF est *homogène en séquence* s'il contient au moins deux séquences et s'il existe i, j tels que $X = C_i^L = C_j^R$. Les clusters correspondant à des ensembles homogènes sont

illustrés par des points verts sur la figure 7.6.

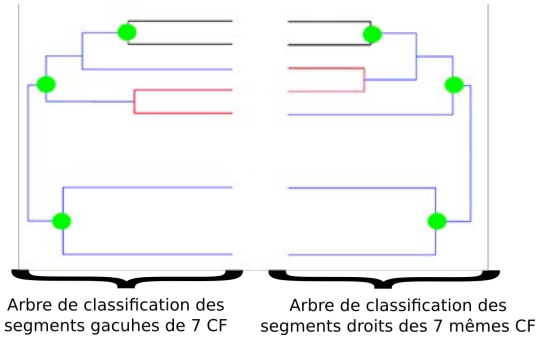

FIG. 7.6: Double classification hiérarchique permettant l'établissement de clusters homogènes (indiqués par les points verts) entre les séquences des segments gauches et celles des segments droits

On ne construit des alignements multiples de séquences que sur les ensembles homogènes en séquence. Les plus proches des feuilles seront plus spécifiques, alors que ceux plus proches de la racine seront plus généraux. Les alignements multiples des séquences dans chaque clusters seront donc de meilleure qualité s'ils correspondent à groupe proche des feuilles. Par contre, ceux-ci aboutiront à des modèles de séquences moins généralistes.

Pour chaque alignement multiple, nous avons construit des HMM séparément pour les séquences des segments gauches et des segments droits. Ces HMM peuvent ensuite scanner des séquences et ainsi détecter des signaux correspondant à des structures de CF.

Nous n'avons pas directement évalué la cohérence et la complétude de ces modèles de séquence pour la détection des structures de CF. Néanmoins, par exemple pour l'identification de séquences capsides dans l'outil VIRALpro (voir section 7.3), nous utiliserons ces détecteurs en ne retenant que les plus performant avec une phase d'apprentissage automatique.

7.3 L'utilisation de la prédiction de CF dans VIRALpro

Les séquences de virus souffrent d'un faible taux d'annotation dans les études métagénomiques [HS13, Sut07, HSS+13]. En effet, leur séquence porte beaucoup de mutations et la similarité de séquence n'est pas suffisante pour identifier systématiquement des homologues connus. Un des critères permettant la reconnaissance et la classification d'un génome viral est la présence d'un gène codant pour une protéine de capside. De plus, l'emplacement dans le génome ainsi que la présence d'autres protéines structurales dans comme les protéines de queue aident à la classification taxonomique. Il est donc nécessaire d'avoir des outils permettant une détection de protéines structurales virales plus puissants qu'une simple détection d'homologues. Un outil appelé iVireons [SAA+12] a

été introduit pour détecter les protéines structurales chez les bactériophages (ou phages, virus infectant les bactéries). Cet outil utilise comme unique information la composition moyenne en acides-aminés de la séquence à identifier. Cette information — calculée sur un jeu de données dit d'apprentissage sur lequel les séquences et leurs annotations sont connues — est dans un premier temps apprise par trois réseaux de neurones afin d'identifier la propension de la séquence à coder pour une protéine de capside, de queue ou plus généralement pour une protéine structurale. En proposant une séquence inconnue à iVireons, les trois réseaux de neurones émettent alors une prédiction quant à la nature de la séquence (protéine de capside, de queue ou structurale de virus). Comme reporté dans [SAA+12], iVireons affiche de bonnes performances pour les phages, mais les résultats se dégradent lorsqu'on essaye de détecter d'autres types de virus.

Durant un séjour de 3 mois à UCI (*University of California in Irvine*), j'ai développé en collaboration avec Pierre Baldi une suite appelée VIRALpro (disponible en ligne `http://scratch.proteomics.ics.uci.edu/`) permettant la détection de séquences de protéines virales de capside (CAPSIDpro) et de queue (TAILpro).

Le principe de détection utilisé dans VIRALpro pourrait très bien s'appliquer à d'autres types de protéines. L'approche générale étant de ne pas utiliser directement l'information d'homologie pour décider si une séquence code pour protéine structurale ou non, mais de penser l'annotation en terme de paradigme séquence-structure-fonction : nous utilisons des informations de séquence pour en déduire de possibles conformations structurales qui, à leur tour, vont fournir des informations permettant de décider si la séquence code pour une protéine de capside (ou de queue). Pour pouvoir prendre en compte ces différentes informations, nous construisons un modèle statistique dont les paramètres sont fixés par une étape d'apprentissage automatique.

Le classificateur statistique retenu est le SVM (*Support Vector Machine*) [CST00] pour sa simplicité d'utilisation ainsi que pour son absence de paramètres de modèle (à la différence par exemple des réseaux de neurones utilisés dans iVireons [SAA+12] où le nombre de neurones ainsi que le nombre de couches doit être optimisé). Le SVM est un classificateur très répandu dans le domaine de l'apprentissage automatique. Les données (ici des séquences) sont décrites par des vecteurs dont les coordonnées sont des *descripteurs* des données (aussi appelés *features* en anglais). Pour des séquences protéiques, un descripteur peut être par exemple le pourcentage en proline (un acide-aminé très conservé en manière générale) dans la séquence. On associe à chaque vecteur (ici à chaque séquence) une classe A ou B (par exemple capside/non capside). Le SVM détermine alors un hyperplan séparant au mieux les vecteurs de classe A et de ceux de classe B en maximisant la marge (*i.e.* la distance[1] entre l'hyperplan et les vecteurs les plus proches).

Nous avons voulu intégrer la détection de CF (voir section 7.2) de protéines de capside à partir de la séquence en tant que descripteur. Cependant, les HMM construits avec la procédure décrite dans 7.2 ne sont pas nécessairement spécifiques des capsides. En effet, d'une part les CF desquels ils sont extraits ne sont pas nécessairement struc-

[1]Ce n'est pas nécessairement la norme issu du produit scalaire standard qui est utilisée. On peut utiliser des noyaux (par exemple un noyau Gaussien $K(x,y) := e^{-\gamma||x-y||^2}$ dans le cas de VIRALpro) afin d'obtenir entre les classes une séparation non-linéaire et plus proche des données d'apprentissage.

turalement spécifiques des capsides. D'autre part, même si le CF est caractéristique de la structure de protéines de capside (tels que ceux présentés dans la section 7.1), les séquences sous-jacentes ne le sont pas nécessairement. On doit donc dans un premier temps "trier" les HMM spécifiques de ceux qui ne le sont pas.

Pour chaque séquence (positive ou négative) i du jeu d'apprentissage (détaillé au paragraphe **Résultats**), chaque HMM h fourni le score $s_{h,i}$ du meilleur *hit* identifié. Pour chaque HMM h on peut ainsi trier les séquences suivant la valeur $s_{h,i}$ et calculer (voir annexe 8) l'AUC ROC AUC_h obtenue sur le jeu d'apprentissage. L'AUC ROC (pour *Area Under Curve ROC* , aire sous la courbe ROC) peut se lire comme la probabilité qu'un exemple positif soit mieux classé qu'un exemple négatif (voir détails et démonstration en annexe 8). Ainsi, plus un HMM est performant pour la classification (*i.e.* plus il est caractéristique des capsides), plus son AUC ROC est élevée.

Même en gardant les HMM les plus spécifiques, si on intègre les scores $s_{h,i}$ directement dans le SVM en tant que descripteurs, on réalise du surapprentissage. En effet, en mesurant l'AUC de classification sur le jeu d'apprentissage (resp. de validation) par le SVM on voit (en figure 7.7) que plus on ajoute de descripteurs $s_{h,i}$, plus la qualité de classification du SVM s'améliore sur le jeu d'apprentissage, mais se détériore simultanément sur le jeu de validation. Ceci signifie que les paramètres appris par le SVM deviennent trop spécifiques du jeu d'apprentissage au détriment de la généralisation à d'autres séquences.

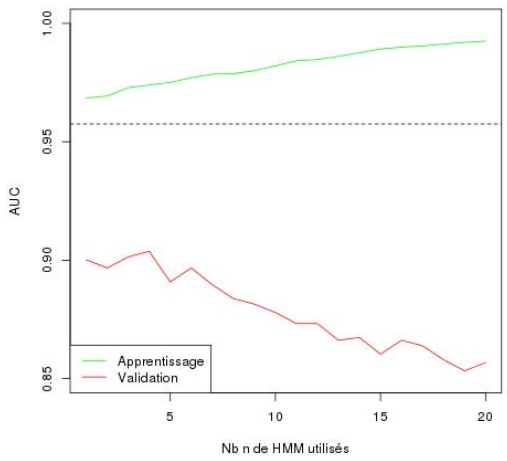

FIG. 7.7: AUC de la classification par le SVM avec comme descripteur la composition en acides-aminés et les scores des des n plus performants HMM (n en abscisse).

Nous avons donc choisi de combiner les scores en une combinaison linéaire via un algorithme de *boosting de préférences* qui consiste à pondérer les HMM suivant les résultats qu'ils fournissent sur le jeu d'apprentissage (les plus spécifiques seront davantage pondérés). On intègre alors uniquement la combinaison linéaire des scores comme des-

cripteur dans le SVM. Ainsi, pour un même nombre de HMM utilisés, le nombre de paramètres total du modèle est fortement réduit par l'utilisation du boosting de préférence, évitant le phénomène de surapprentissage. Le trait en pointillé en figure 7.7 représente l'AUC de la classification sur le jeu de validation en utilisant comme descripteurs la composition moyenne en acides-aminés ainsi que le *boosting de préférence*. On voit nettement l'intérêt de ce boosting par rapport à l'utilisation immédiate des scores des HMM comme descripteurs.

Pour réaliser ce boosting, nous avons utilisé un algorithme décrit dans [FISS03] qui assure à chaque itération de l'algorithme la diminution d'une borne majorant l'erreur de classification sur le jeu d'apprentissage.

Formellement, on dispose d'un ensemble d'objets X (dans notre cas des séquences) et chaque objet x possède une classe $c(x)$ valant 0 ou 1 (par exemple dans le cas des capsides, 1 est associé aux séquences de capside et 0 aux autres). Nous appellerons *évaluateur faible sur un ensemble d'objets* X une fonction $e : X \to \mathbb{R}_{>0}$. Un évaluateur faible attribut une note pour chaque objet : les notes les plus élevées correspondent à la classe 1, les plus faibles à la classe 0. Cependant, un évaluateur faible peut se tromper et donner des notes hautes à des objets de classe 0, et inversement. C'est pourquoi le boosting de préférence permet de construire un évaluateur dit *fort* qui, à partir d'une combinaison linéaire des notes de plusieurs évaluateurs faible, attribut de manière plus fiable une meilleure note aux objets de classe 1 qu'aux objets de classe 0. Ainsi, plus les évaluateurs faibles sont complémentaires (*i.e.* sont justes sur une sous-partie des objets), plus l'évaluateur fort sera performant.

Supposons qu'on dispose de N évaluateurs faibles $e_0, ..., e_{N-1}$ (dans notre cas le score des hits des HMM). On note R la fonction de rang :

$$R(a,b) := \mathbf{1}_{c(a)\leq c(b)} \tag{7.1}$$

On définit alors l'erreur d'un évaluateur e comme : $\epsilon(e) = \sum_{a,b\in X} R(a,b)\mathbf{1}_{e(b)\leq e(a)}$.

On définit récursivement les fonctions de rang R_i :

$$\begin{cases} R_0(a,b) & := & \dfrac{R(a,b)}{Z_0} \\ Z_0 & := & \sum\limits_{a,b\in X} R(a,b) \end{cases}$$

$$\begin{cases} R_{i+1}(a,b) & := & \dfrac{R_i(a,b)e^{\alpha_i[e_i(a)-e_i(b)]}}{Z_i} \\ Z_i & := & \sum\limits_{a,b\in X} R_i(a,b)e^{\alpha_i[e_i(a)-e_i(b)]} \end{cases}$$

Avec : $\alpha_i := \dfrac{1}{2}\ln\dfrac{1-k_i}{1+k_i}$ où $k_i := \sum\limits_{a,b\in X} R_i(a,b)[e_i(a)-e_i(b)]$.

On construit à chaque itération alors un évaluateur fort $E_n(x) := \sum\limits_{i=0}^{n-1} \alpha_i e_i(x)$. A chaque itération, si l'évaluateur faible e_i est faux (c'est-à-dire qu'il accorde de bonnes notes aux séquences de classe 0, et de mauvaises à celles de classe 1), on aura k_i tendant vers 1, et donc un α_i tendant vers $-\infty$ (*i.e.* une pondération négative de e_i dans

l'évaluateur fort). Inversement, si e_i est précis, alors $k_i < 0$ et la pondération α_i sera strictement positive.

On peut montrer [FISS03] que l'erreur $\epsilon(E_n)$ commise par l'évaluateur fort E_n est majorée par $\prod_{i=0}^{n-1} Z_i$. Comme par ailleurs on montre que $Z_i \leq \sqrt{1-k_i^2} \leq 1$, cette borne supérieure décroit à chaque itération de l'algorithme : $\forall n < N, \epsilon(E_n) \leq \epsilon(E_{n+1})$.

On obtient ainsi une combinaison linéaire E_N des hits des HMM qu'on intègre en tant que descripteur dans le SVM.

Finalement, pour entraîner le SVM utilisé dans VIRALpro, nous avons utilisé un ensemble de 26 descripteurs (*features*) :

Composition Comme dans iVireons, nous avons utilisé la composition moyenne en acides-aminés comme information globale de séquence : 20 *features*.

Structures secondaires Nous avons ensuite intégré une information de prédiction de structure : le taux moyen prédit par SSpro [MB14] de structures secondaires : 3 *features* (hélice α, brin β, ou sans structure secondaire).

CF Nous avons utilisé les HMMs décrit dans la section 7.2 comme détecteur faible de CF de capside que nous avons transformés en détecteur fort via une étape de *boosting* décrite dans [FISS03] et détaillée ci-après : 3 *features* (combinaison linéaire *boostée* des scores de HMM, meilleure *e*-value parmi les HMM et nombre de *hits* dont la *e*-value est inférieure à 10).

Résultats Les jeux de données utilisés pour l'apprentissage et la validation sont décrits en annexe 9. Afin d'évaluer les performances de VIRALpro, on réalise une validation croisée en 10 échantillons : le jeu d'apprentissage est découpé en 10 parties équilibrées en ratio négatif-positif aléatoirement choisies, et une des 10 parties est successivement choisie comme étant l'échantillon de test, le SVM et le boosting de préférences étant entraînés sur les 9 restantes. Pour chaque séquence testée, le SVM affiche un score qui permet de classer les séquences de l'échantillon de test en fonction de leur propension à être une capside (dans le cas de CAPSIDpro) ou une protéine de queue (dans le cas de TAILpro).

Le taux d'exemples négatifs et positifs dans nos jeux de données étant du même ordre de grandeur, on utilise des courbes ROC (pour *Receiver Operating Characteristic*) pour rendre compte des résultats. Par ailleurs, nous avons gardé la dernière "passe" de la validation croisée afin de montrer les courbes ROC obtenues. Le jeu de test correspondant à cette passe sera appelé jeu de validation.

La figure 7.3 montre les courbes ROC pour la détection de capsides et de queues par VIRALpro en comparaison de iVireons. On voit que même sur le jeu de données restreint aux phages, VIRALpro montre une amélioration substantielle de la détection. Cette amélioration est particulièrement prononcée dans le cas des séquences difficiles (seules les séquences du jeu de validation non-homologues avec le jeu d'apprentissage — *e*-value ≤ 0.001 — sont gardées) où BLAST et iVireons affichent des performances proches de la classification aléatoire (AUC respectives de 52.8% et 60.4%).

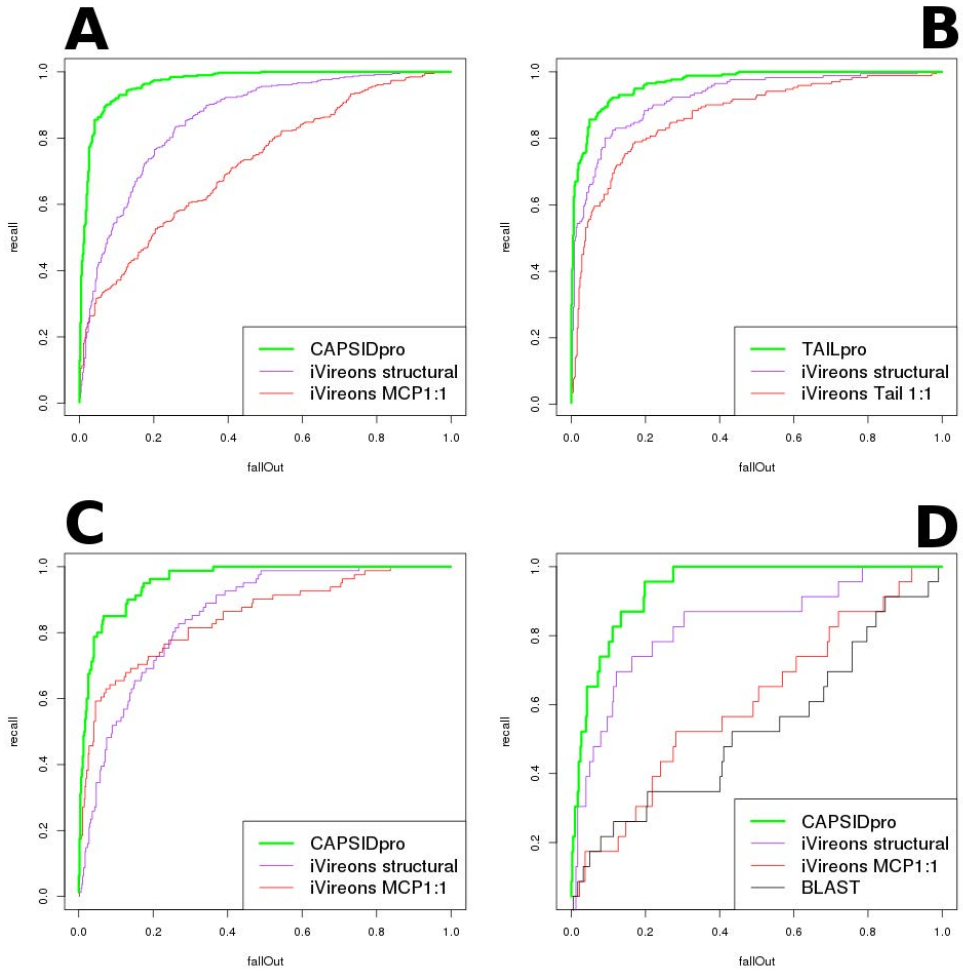

FIG. 7.8: Courbes ROC sur le jeu de validation de CAPSIDpro (A) et de TAILpro (B), en ne gardant que les séquences de bactériophages dans les exemples positifs du jeu de validation (C) et en ne conservant que les séquences du jeu de validation ne partageant aucune homologie (e-value ≥ 0.001 avec le jeu d'apprentissage) (D).

Le tableau 7.1 montre les statistiques des résultats pour la validation croisée ainsi que sur le jeu de validation :

Test	Measure	iVireons MCP 1 :1	iVireons Structural	CAPSIDpro
iV_{test}	Accur.	90%	80%	**97.3%***
10-fold iV_{capsid}	Accur.	91.3%	-	**96.8%*** ±2.5
10-fold $Cpro_{train}$	AUC	74.1%±1.6	85.5%±0.9	**96.1%** ±0.6
	F-Meas.	51.1%±2.7	78.5%±1.3	**89.7%** ±1.2
Validation Set	AUC	72.2%	85.7%	**96.9%**†
	F-Meas.	47.8%	77.9%	**91.0%**†

TAB. 7.1: Résultats de iVireons et CAPSIDpro. Afin d'établir une comparaison juste, lorsque les résultats sont marqués d'une *, CAPSIDpro a été entraîné avec iV_{capsid} , aussi proche que possible du jeu d'apprentissage de iVireons MCP 1 :1 [SAA+12] (la manière exacte de découper les échantillons d'apprentissage pour la validation croisée de iVireons ne sont plus disponibles).
Lorsque les tests portent sur le jeu de validation, CAPSIDpro a été entraîné avec les 90% restant du jeu d'apprentissage (marqué d'un †).

Les descripteurs liés à la détection des CF de capsides dans les séquences ont une influence plus prononcée pour des séquences n'ayant aucune similarité avec les séquences du jeu d'apprentissage. En effet, lorsque nous entraînons le SVM *sans* les descripteurs liés à la détection des CF, sur la validation croisée, l'AUC baisse en moyenne de 0.4%, mais l'AUC baisse de 2% sur le jeu de validation duquel on retire tous les homologues (*i.e.* le même jeu de données qu'en figure 7.8 D).

Test	Mesure	iVireons Tail 1 :1	iVireons Structural	TAILpro
iV_{test}	Accur.	80%	77%	**89.9%***
10-fold iV_{tail}	Accur.	79.9%	-	**89.4%*** ± 2.4
10-fold $Tpro_{train}$	AUC	68.1%±2.9	91.2%±1.3	**96.8%** ±0.4
	F-Meas.	27.0%±3.4	83.8%±1.7	**91.0%** ±1.0
Validation Set	AUC	87.0%	92.3%	**96.7%**†
	F-Meas.	80.5%	83.7%	**89.8%**†

TAB. 7.2: Résultats de iVireons et de TAILpro. Mêmes notations et remarques que dans le tableau précédent, transposées au cas de TAILpro et iVireons tail 1 :1.

Enfin, nous avons scanné avec VIRALpro les séquences provenant des études métagénomiques marines suivantes :

RnaCoastal Virus à ARN provenant des eaux littorales en Colombie Britanique [CLS06],

Oresund-Struct, Oresund-Hypo, et Oresund-NonStruct phages provenant du détroit d'Öresund en mer Baltique [HSS+13],

Moore phages séquencés par le Broad Institute, sous le projet "Marine Phage, Virus, and Virome Sequencing" de la Gordon and Betty Moore's Foundation.

Dans tous les jeu de données — détaillés en annexe —, nous avons gardé seulement les séquences dont l'annotation n'avait pas pu être déterminée par les auteurs. Ainsi, les séquences résultantes ne partagent qu'une très faible homologie (0% des séquences pour RnaCoastal et de Oresund-Struct, moins de 2% pour les autres) avec les séquences du jeu d'apprentissage de VIRALpro.

Le jeu de données des phages du détroit d'Öresund vient en trois catégories : Oresund-Struct, Oresund-Hypo, et Oresund-NonStruct. Le premier est l'ensemble des séquences qui ont été identifiées par spectrométrie de masse, le second est un ensemble de séquences hypothétiques, et le dernier est un ensemble de séquences qui n'ont pas été observées par spectrométrie de masse et qui sont supposées être pour la plupart des séquences de protéines non structurales. Cependant, grâce à un échange avec les auteurs nous savons que des protéines structurales sont tout de même contenues dans ce jeu, puisque notamment les protéines de petite taille échappent nécessairement à la spectrométrie de masse. Ceci peut expliquer le petit pourcentage de séquences de ce jeu détectées par VIRALpro comme structurales.

Jeu	Rappel CAPSIDpro	Rappel TAILpro
RnaCoastal (86)	32.6%	22.1%
Oresund-Struct (85)	32.9%	45.9%
Oresund-Hypo (524)	14.5%	9.2%
Oresund-NonStruct (156)	9.0%	7.7%
Moore (1172)	26.4%	24.9%

TAB. 7.3: Rappel obtenu avec VIRALpro sur les cas réels. Les nombres indiqués entre parenthèses représentent le nombre de séquences de chacun des jeux de données.

Il est intéressant de remarquer qu'on détecte un certain nombre de protéines de queue dans le jeu RnaCoastal, laissant penser qu'il pourrait s'agir de protéines de fibre, ce qui confirmerait la présence de rotavirus comme suggéré dans l'étude originale [CLS06].

7.4 Conclusion et perspectives

Nous avons basé l'application des CF sur l'identification de protéines de virus dont la caractérisation tant structurale que séquentielle est très parcellaire dans les études précédemment publiées. Nous avons vu dans un premier temps comment on pouvait identifier des CF caractéristiques de structures de capsides. Nous avons obtenu une bonne caractérisation de quelques sous-groupes de capsides de virus. Ceci pourrait permettre l'annotation automatique de nouvelles structures en identifiant dans celles-ci les CF spécifiques identifiés. Cependant, il est fort probable que pour d'autres types de capsides, aucun CF ne soit individuellement spécifique, mais que la présence simultanée dans la structure de plusieurs CF permette leur caractérisation. Il serait alors intéressant d'utiliser des outils symboliques tels que les concepts formels ou statistiques tel que

les réseaux bayésiens pour combiner l'occurrence structurale des CF et ainsi obtenir une caractérisation plus complète des structures de capsides de virus.

Nous avons montré en section 6.2.2 que deux CF fusionnent pour un τ suffisamment élevé. Il se peut alors que dans une protéine, deux CF soient à la limite de fusionner et que dans une protéine similaire ceux-ci ne forment qu'un seul grand CF. Notre méthode actuelle d'identification de CF commun à une famille ne permet pas de reconnaître dans ce cas qu'il existe une sous-structure commune de CF. Une possibilité serait alors d'aligner dans un premier temps les contacts de différentes protéines (comme cela peut être fait avec le *Contact Map Overlap* [AMDY11]), puis d'extraire simultanément les CF autour d'un contact en prenant l'intersection des rectangles entourant les composantes connexes dans les différentes protéines. On obtiendrait alors un ensemble de structures de tailles identiques, partagé par l'ensemble des structures alignées, que l'on pourrait qualifier de *motif de contact*. Ces motifs de contact seraient alors partagés par plus de membres de la famille que les CF pris individuellement dans chaque protéine, et permettraient peut-être une meilleure caractérisation structurale de familles protéiques.

Au niveau de la séquence, nous avons présenté une méthode permettant la détection depuis la séquence de CF partagés par plusieurs protéines d'une même famille. Cette méthode de détection a été intégrée avec des techniques d'apprentissage automatique dans le développement de l'outil VIRALpro permettant la reconnaissance de séquences de protéines de capsides et de queue chez les virus. Nous avons finalement observé que VIRALpro affichait des résultats significativement plus performants que ceux de l'état de l'art en matière de détection de ces protéines virales.

Il reste plusieurs pistes d'amélioration de l'outil VIRALpro. Le regroupement des CF par découpage taxonomique d'une classification hiérarchique a été choisi car les performances étaient meilleures sur une étude préliminaire qu'un regroupement avec une méthode de type *k-means*. Nous pourrions peut-être améliorer la similarité structurale intra-cluster en validant *a posteriori* la qualité de chaque cluster par un seuil de similarité structurale. Concernant la détection de CF, nous construisons les HMM à partir des séquences associées aux structures contenues dans les clusters. Nous pourrions utiliser des outils plus puissants comme PSI-BLAST pour construire un ou plusieurs profils par cluster qui seraient sûrement plus performants. Aussi, nous n'avons pas évalué la cohérence (au sens du chapitre 4) des HMM par rapport à la structure qu'ils décrivent. Plutôt que de réaliser l'apprentissage par rapport à leur capacité à identifier des protéines de capside ou de queue, un premier tri pourrait être effectué en ne conservant que les modèles les plus cohérents.

De plus, lorsque nous détectons les CF depuis la séquence, ceux-ci sont détectés indépendamment, et la co-occurrence de CF n'est pas prise en considération. Il pourrait être intéressant de poursuivre la recherche de caractérisation de famille protéique depuis la séquence en prenant en compte cette co-occurrence de CF (en utilisant par exemple un réseau bayésien), voire leur séquentialité (grammaire permettant de générer les différentes imbrications de CF d'une même famille).

Conclusion

Cette thèse vise à l'élaboration de techniques plus poussées que la simple comparaison de séquence et de structure pour l'identification de protéines. Pour cela, nous avons focalisé notre étude sur la caractérisation au niveau local et notamment sur l'identification depuis la séquence de structures locales conservées.

Résumé des contributions

Motivés initialement par le lien séquence-structure dans les protéines, nous avons été amenés à formaliser la notion de *prédictibilité* d'une structure (chapitre 4). Nous avons formulé deux hypothèses pouvant aboutir à des motifs structuraux prédictibles : la localité spatiale ainsi que le voisinage séquentiel renforcent le caractère prédictible d'un motif. Ces hypothèses nous ont amené à définir (chapitre 6) les *fragments en contacts* (ou CF pour *contact fragments*).

Pour pouvoir structuralement comparer les CF dont les extrémités peuvent légèrement varier d'une protéine à l'autre, les scores de comparaison structurale usuels nécessitent une phase d'alignement des acides-aminés à comparer, nécessitant l'introduction de seuils arbitraires et réduisant la qualité de classification par le non-respect de l'inégalité triangulaire. Pour concilier la tolérance aux indels et le respect de l'inégalité triangulaire, nous avons introduit une nouvelle mesure de dissimilarité structurale : l'ASD (chapitre 5). Celle-ci se base sur une comparaison globale des fragments via la comparaison du spectre (*i.e.* les modules des coefficients de Fourier) des matrices des distances internes entre atomes de carbone C_α. Nous avons montré que cette mesure était également pertinente (notamment affichant des performances nettement meilleures que le RMSD, le TM-Score et le BC score) pour la comparaison de fragments simples (*i.e.* contigus en séquence) sur quelques exemples pratiques (section 5.5) de fouille structurale mais aussi de classification non supervisée.

Pouvant alors extraire, comparer et regrouper les CF, nous avons montré (section 6.4) dans un premier temps qu'ils étaient davantage prédictibles que les fragments simples et que les paires de fragments qui ne sont pas en contact. Nous avons ensuite présenté deux exemples principaux de caractérisation de protéines à l'aide des CF : l'un au niveau de la structure avec la découverte de CF spécifiques aux capsides (section 7.1), et l'autre au niveau de la séquence (section 7.3) par le développement de VIRALpro (disponible en ligne `http://scratch.proteomics.ics.uci.edu/`) permettant l'identification de protéines structurales de virus dans des séquences protéiques.

Pour résumer, notre contribution se décline selon trois apports principaux : l'ASD pour la comparaison de fragments de structure, les CF comme des descripteurs prédictibles de structures et leur apprentissage automatique pour l'identification de protéines de virus.

Résumé des perspectives

Bien que l'ASD et les CF permettent déjà d'obtenir des résultats, de nombreuses perspectives restent ouvertes.

D'une part, des pistes restent à explorer dans l'étude de l'ASD. Actuellement, l'ASD compare deux-à-deux les coefficients de Fourier, mais on pourrait envisager de repérer au sein d'une famille structurale quelle part du spectre est commune à tous les membres de la famille. Il serait alors possible de créer une signature de la famille sans avoir recours à un alignement structural algorithmiquement coûteux de toutes les structures. Par ailleurs, lors de leur comparaison avec l'ASD, les coefficients de Fourier ne sont pas pondérés. On pourrait éventuellement obtenir des comparaisons plus fines en pondérant les coefficients selon leur capacité à discriminer les structures. Enfin, l'ASD n'a été testée que pour le cas de fragments de structure où les indels restent relativement courts. Il serait intéressant d'évaluer sa capacité à comparer des protéines entières comportant des insertions et délétions de grandes portions de structure. On pourrait notamment observer s'il persiste une trace de la structure initiale dans le spectre de la matrice des distances internes.

Concernant les CF, l'état actuel de leur étude permet de les situer comme des descripteurs de structure permettant une caractérisation structurale et séquentielle des protéines. Parmi l'ensemble des CF d'une protéine, il existe couramment des CF qui ne sont pas porteurs d'information caractéristique de la famille de la protéine. Pour les caractérisations structurales et séquentielles que nous avons présenté, l'identification de chaque CF est indépendante. La combinatoire des occurrences de CF dans une protéine pourrait être intégrée dans des méthodes d'apprentissage statistique comme les SVM ou réseaux de neurones, mais aussi dans des méthodes symboliques comme l'analyse de concepts formels où chaque CF serait un attribut d'une structure. La détection simultanée de CF pourrait notamment être intégrée dans un cadre formel utilisant des outils mathématiques structuraux tels que les faisceaux (section 4.3), ce qui permettrait de gérer systématiquement la détection chevauchante de CF mutuellement exclusifs. La difficulté principale à laquelle nous avons été confronté dans ce cadre est le sur-apprentissage : pour une famille de protéines donnée, l'apprentissage des CF caractéristiques était souvent trop spécifique au jeu d'apprentissage et prévenait toute généralisation. Ce sur-apprentissage pourrait venir de la grande hétérogénéité des structures de capsides, et il serait intéressant de le comparer à une étude de caractérisation de famille structurale plus homogène (protéines de même fold par exemple).

La séquentialité ainsi que la hiérarchie formée par l'ensemble des CF d'une protéine pourraient également contribuer à une meilleure caractérisation de familles protéiques. On pourrait par exemple identifier une grammaire qui dériverait toutes les imbrications

possibles de CF pour une famille donnée de protéines.

Plusieurs extensions de la définition même de CF pourraient être pertinentes. D'une part vers un élargissement de la portée des CF, la co-occurrence locale de CF permettrait par exemple l'extension de la définition de CF à des super-CF : de multiples fragments en contact par analogie avec les super-structures secondaires. En effet, il se peut qu'aucun CF ne soit individuellement caractéristique d'une famille de protéine (et même que chacun d'entre eux soit individuellement ubiquitaire), mais que leur ensemble forme un super-CF permettant de définir une signature structurale. D'autre part une restriction de la portée des CF pourrait être intéressante. En effet, au sein d'une famille, il se peut que seulement une sous-partie d'un CF soit commune à toutes les protéines. Les méthodes que nous avons développées actuellement ne permettent pas l'identification de tels "sous-CF". On pourrait définir des *motifs de contact* comme étant la partie commune de fragments en contact similaires dans des structures d'une même famille. De tels motifs pourraient être définis à partir de l'alignement multiple de contact dans des structures tel que suggéré dans la section 7.4, dont la mise en œuvre pourrait néanmoins s'avérer difficile avec un grand nombre de protéines.

Nous avons illustré l'utilisation des CF pour la caractérisation de familles de protéines. Un autre problème exposé dans le chapitre 4, et que nous n'avons pas exploré, est la définition d'une couverture de l'ensemble des protéines composée de structures prédictibles. Les CF semblent être de bons candidats pour une telle couverture : ils sont définissables universellement à partir de toute structure de protéine (on peut les extraire directement à partir des coordonnées spatiales des atomes de la protéine) et leur classification à grande échelle permettrait d'établir une librairie pouvant être vue comme un alphabet structural à fragment double. Les éléments de cette librairie devraient alors bénéficier d'une bonne prédictibilité d'après les résultats de l'expérience en section 6.4. La prédictibilité des CF pourrait par ailleurs être améliorée en utilisant des modèles plus fins de séquence qui prendraient par exemple en compte les corrélations entre acides-aminés en interaction. De tels modèles de séquence pourraient être définis grâce à des outils statistiques comme les *Markov Random Field* dont la dépendance statistique serait calquée sur la proximité spatiale ou encore symboliques tels que le modèle logique présenté en section 6.5.

D'un point de vue plus général, l'expérience présentée en section 6.4 montre d'une part que les CF sont davantage prédictibles que des paires de fragments c'est-à-dire que pour des structures en contact, la conservation de structure sera plus forte pour une même similarité de séquence. Dans les méthodes actuelles d'évaluation de similarité de séquence, le même poids est attribué à une mutation intervenant au niveau d'une boucle que dans une structure secondaire enfouie à l'intérieur de la protéine. Lorsque la similarité de séquence est utilisée afin de rendre compte de la similarité de structure — et donc de la similarité de fonction — il serait plus judicieux d'attribuer un poids plus fort à une mutation intervenant dans une structure libre que dans une structure en contact. On pourrait alors prédire les parties enfouies (en donc en contact) des protéines afin de générer une pondération pour les scores de substitution.

Troisième partie

Annexes

Chapitre 8

Définitions

Il existe plusieurs moyens d'évaluer les performances d'un score par rapport à sa capacité à différencier des objets de classes différentes. Nous présentons ici la courbe ROC et la courbe précision-rappel.

Commençons par définir X l'ensemble des données, dont chaque élément $x \in X$ possède une classe c_x valant 0 ou 1, et soit $F_\gamma : X \to \{0, 1\}$ le classificateur de paramètre γ. Par exemple lorsqu'on dispose d'un score $s(x)$ associé à chaque objet x, les classificateurs associés sont généralement :

$$F_\gamma(x) := \begin{cases} 0 & \text{si} \quad s(x) \leq \gamma \\ 1 & \text{sinon} \end{cases} \tag{8.1}$$

Un tel score peut être la distance (signée) à la frontière de décision dans le cas d'un SVM (comme dans VIRALpro par exemple), ou la similarité de l'élément x à un élément de référence (comme dans l'expérience de recherche des structures de ZF dans la section 5.5.1).

On définit alors les termes suivants attribués à chaque élément x suivant sa classe c_x et le résultat du classificateur $F_\gamma(x)$:

	$c_x = 0$	$c_x = 1$
$F_\gamma(x) = 0$	Vrai négatif (VN)	Faux négatif (FN)
$F_\gamma(x) = 1$	Faux positif (FP)	Vrai positif (VP)

Un classificateur donnant un taux de 100% de vrais positifs ainsi que de vrais négatifs est un classificateur "parfait" car la connaissance de la valeur du classificateur indique directement la classe de l'élément.

On peut évaluer la qualité d'un classificateur avec les ratios suivants :

$$\text{Rappel} = \frac{VP}{VP + FN}$$

$$\text{Précision} = \frac{VP}{VP + FP}$$

$$\text{Spécificité} = \frac{VN}{VN + FP}$$

Courbe ROC Pour évaluer une famille de classificateurs F_γ en fonction du paramètre γ, on utilise une courbe ROC (pour *Reciever Operating Characteristic*) où chaque point de la courbe correspond à une valeur γ, dont l'ordonnée est le rappel de F_γ et l'abscisse $1 -$ Spécificité de F_γ.

Si la courbe ROC est fortement courbée vers le coin de coordonnées $(0, 1)$, cela signifie qu'il y a une valeur de γ pour laquelle on peut avoir un rappel proche de 1 pour une spécificité proche de 1 également, c'est-à-dire qu'on dispose d'un classificateur de bonne qualité.

Inversement, si le classificateur F_γ est aléatoire, suivant une loi de Bernoulli de paramètre γ alors, en définissant A comme le nombre d'éléments de classe 1 et, $M := |X|$ comme le nombre total d'éléments, on observe que les espérances des valeurs de spécificité et de rappel sont directement liées :

$$\text{Rappel} = \frac{VP}{VP + FN} = \frac{\gamma A}{\gamma A + (1-\gamma)A} = \gamma \tag{8.2}$$

$$\text{Spécificité} = \frac{VN}{VN + FP} = \frac{(1-\gamma)(1-A)}{(1-\gamma)(1-A) + \gamma(1-A)} = 1 - \gamma \tag{8.3}$$

Dans le cas d'un classificateur aléatoire, on a alors pour tout γ : Rappel $= 1 -$ Spécificité et la courbe ROC suit alors la première diagonale.

L'aire sous la courbe ROC, également notée AUC ROC (pour *Area Under Curve ROC*) ou par abus de notation AUC, est donc proche de 0.5 dans le cas d'un classificateur aléatoire, et proche de 1 pour un classificateur de bonne qualité.

Proposition 8.0.1. *Sous certaines conditions de régularité, lorsque le paramètre γ correspond à un seuil sur un score $s(x)$, l'AUC ROC correspond à la probabilité que le score soit plus élevé pour un élément de classe* 1 *que pour un élément de classe* 0*.*

Démonstration. Soient $f(\gamma) := P(s(x) \leq \gamma | c(x) = 0)$ (*i.e.* la Spécificité de F_γ) et $g(\gamma) := P(s(x) \geq \gamma | c(x) = 1)$ (*i.e.* le Rappel de F_γ). On supposera que f et g sont dérivables, que f' ne s'annule pas et que f est inversible.

La probabilité que le score soit plus élevé pour un élément y de classe 1 que pour un élément x de classe 0 est : $p := P(s(x) \leq s(y) | c(x) = 0 \wedge c(y) = 1) = \int_{\mathbb{R}} f'(\gamma).g(\gamma)d\gamma$ (ou de manière équivalente : $p = -\int_{\mathbb{R}} f(\gamma).g'(\gamma)d\gamma$, résultat trivial par intégration par parties).

En posant $\gamma = f^{-1}(x)$, alors $d\gamma = \dfrac{dx}{f'(f^{-1}(x))}$ et on obtient :

$$\begin{aligned} p &= \int_0^1 f'(f^{-1}(x)) g(f^{-1}(x)) \frac{dx}{f'(f^{-1}(x))} \\ &= \int_0^1 g(f^{-1}(x))dx \end{aligned} \tag{8.4}$$

En posant $y = 1 - x$:

$$\begin{aligned} p &= \int_0^1 g(f^{-1}(1-y))dy \\ &= \int_0^1 g((1-f)^{-1}(y))dy \end{aligned} \tag{8.5}$$

La valeur de p correspond alors à l'aire sous la courbe définie par l'ensemble des points $\{(1 - f(\gamma), g(\gamma))|\gamma \in \mathbb{R}\}$, qui est la courbe ROC de F_γ. □

Courbe Précision-Rappel Si dans un jeu de données, il existe un grand nombre d'éléments de classe 0 par rapport au nombre d'éléments de classe 1 (comme dans l'expérience de recherche de ZF en section 5.5.1, alors pour deux classificateurs de qualité honnête, le nombre de faux positifs sera négligeable par rapport au nombre de vrai négatifs sur l'ensemble du jeu de données X. Ainsi, leur spécificité sera pour chacun excellente : Spécificité $= \frac{VN}{VN+FP} \approx \frac{VN}{VN} = 1$ et il sera difficile de les différencier en utilisant une courbe ROC. Dans ce cas, on privilégie l'utilisation d'une courbe précision-rappel (PR) donnant une meilleure discrimination entre les bons classificateurs.

Chapitre 9

Jeux de données

SkF

Pour les tâches nécessitant d'évaluer la comparaison un-contre-un de fragments de protéines (pour évaluer une distribution de score par exemple), nous avions besoin d'un jeu de données relativement réduit pour des raisons de temps de calcul. A partir des 40 domaines protéiques du jeu de données classique de Skolnick [LCWI01], nous avons extrait tous les fragments (avec chevauchement) des longueur N, avec N prenant les valeurs $20, 30, 40, 50$ et 60. Nous avons noté $\mathbf{SkF}_N$ chacun de ces jeux de données. Les identifiants des domaines Astral utilisés pour l'extraction des fragments de SkF_N sont les suivants : d1amk_, d1aw2A, d1b00A, d1b71A, d1b9bA, d1bawA, d1bcfA, d1btmA d1byoA, d1byoB, d1dbwA, d1dpsA, d1fha_, d1htiA, d1ier_, d1kdi_ d1nat_, d1nin_, d1ntr_, d1pla_, d1qmpA, d1qmpB, d1qmpC, d1qmpD d1rcd_, d1rn1A, d1rn1B, d1rn1C, d1tmhA, d1treA, d1tri_, d1ydvA d2b3iA, d2pcy_, d2plt_, d3chy_, d3ypiA, d4tmyA, d4tmyB, d8timA.

ZF

Pour l'expérience d'identification structurale de fragments de Zinc Finger (ZF), nous avons utilisé les identifiants PDB listé dans le référencement croisé du motif Prosite de ZF C2H2 C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H dont l'identifiant est *PS00028* dans la base Prosite (Release 20.99 [SCC+13b]).

Les instances de ce motif de ZF ont des longueurs variables à cause des différents sites d'insertion. Afin de pouvoir utiliser les outils de comparaison structurale qui sont limités aux structures de taille identique, nous avons systématiquement extrait une fenêtre de 23 résidus débutant au premier acide-aminé identifié comme appartenant au motif (au premier C). Quand plusieurs modèles était présent dans les fichiers PDB correspondant, nous avons utilisé uniquement celui qui venait en premier dans le fichier. Nous avons également retiré le fragment correspondant aux acides-aminés 18 à 41 du PDB 2MA7 car sa structure linéaire indiquait clairement qu'il ne s'agissait pas d'une structure de Zinc Finger.

Le jeu de données résultant est dénoté **ZF**, et les identifiants ainsi que les positions des motifs sont les suivants :

PDB	Pos.	PDB	Pos.	PDB	Pos.	PDB	Pos.	PDB	Pos.	PDB	Pos.	PDB	Pos.
1A1F	137	1UBD	327	2EM3	15	2EPR	357	2YT9	392	1ZAA	37	2I13	108
1A1H	107	1UBD	355	2EM5	15	2EPS	415	2YT9	422	1ZAA	7	2I13	24
1A1H	137	1UBD	385	2EM6	15	2EPT	79	2YTA	141	2CSH	40	2I13	80
1A1H	165	1UN6	107	2EM8	15	2EPU	107	2YTB	198	2DLK	41	2JP9	69
1A1I	107	1UN6	137	2EM9	15	2EPV	810	2YTD	15	2DLQ	97	2JP9	9
1A1I	137	1VA3	599	2EMB	15	2EPW	920	2YTF	15	2DMI	22	2JP9	97
1A1J	107	1WIR	18	2EMC	15	2EPY	530	2YTG	15	2DRP	113	2JPA	69
1A1K	107	1X5W	12	2EME	15	2EPZ	507	2YTH	15	2EBT	405	2JPA	9
1A1K	137	1X6E	17	2EMF	15	2EQ0	459	2YTJ	15	2EE8	48	2JPA	97
1A1K	165	1X6E	45	2EMG	15	2EQ1	487	2YTK	15	2EL6	15	2LCE	48
1A1L	107	1X6F	28	2EMK	15	2EQ3	711	2YTO	15	2ELO	12	2LT7	496
1A1L	137	1X6H	18	2EML	15	2EQ4	458	2YTP	15	2ELS	12	2LT7	524
1AAY	137	1XF7	5	2EMP	15	2EQW	414	2YTQ	15	2ELY	15	2LT7	552
1AAY	165	1YUJ	36	2EMW	13	2GLI	106	2YTR	15	2ELZ	15	2LV2	31
1ARD	106	1ZAA	65	2EMX	13	2GLI	202	2YTT	15	2EM0	15	2LVT	32
1ARF	106	1ZFD	44	2EMY	15	2HGH	137	2YU5	15	2EM1	13	2M0E	34
1BBO	32	1ZNF	3	2EN1	15	2I13	52	2YU8	15	2EM2	15	2MDG	6
1BHI	9	1ZR9	45	2EN2	14	2J7J	34	4F2J	473	2EM4	15	2PRT	355
1EJ6	183	2ADR	106	2EN4	15	2J7J	4	4F6M	524	2EM7	15	2PRT	385
1G2D	107	2ADR	134	2EN6	15	2JP9	39	4F6M	552	2EMA	15	2PRT	413
1G2D	137	2COT	21	2EN7	15	2JPA	39	7ZNF	5	2EMH	15	2RPC	123
1G2D	165	2COT	49	2EN8	15	2KMK	32	1A1G	107	2EMI	15	2RPC	93
1G2D	207	2CSE	183	2EN9	15	2KMK	4	1A1G	137	2EMJ	15	2RSH	12
1G2D	237	2CSE	51	2ENC	15	2KMK	60	1A1I	165	2EMM	15	2RSJ	67
1G2F	107	2CT1	18	2ENE	15	2KVF	6	1A1J	137	2EMV	15	2WBS	432
1G2F	207	2CT1	48	2ENF	15	2L1O	8	1A1L	165	2EMZ	15	2WBT	103
1G2F	237	2CTD	65	2ENH	15	2LCE	20	1AAY	107	2EN0	13	2WBT	77
1JK1	107	2D9H	10	2EOE	15	2LV2	59	1ARE	106	2EN3	15	2YRJ	15
1JK1	137	2D9H	41	2EOF	15	2LVR	6	1G2D	265	2ENA	15	2YSP	15
1JK2	137	2DLK	10	2EOG	13	2M0D	6	1G2F	137	2ENT	353	2YT9	364
1JK2	165	2DLQ	10	2EOI	13	2M0F	62	1G2F	165	2EOH	15	2YTE	13
1LLM	106	2DLQ	69	2EOJ	15	2M9A	16	1G2F	265	2EOM	15	2YTI	15
1LLM	206	2DMD	11	2EOK	13	2M9A	44	1JK1	165	2EOS	14	2YTM	15
1P47	107	2DMI	83	2EOL	13	2M9A	74	1JN7	11	2EOV	15	2YTN	15
1P47	137	2DRP	143	2EON	15	2MA7	46	1NCS	34	2EOW	15	2YTS	15
1PAA	134	2EBT	375	2EOO	15	2PRT	325	1NJQ	8	2EOZ	15	3AX1	500
1SP1	5	2EBT	435	2EOP	15	2RSI	39	1P7A	14	2EP0	15	3MJH	43
1SP2	5	2EE8	20	2EOQ	15	2RSI	67	1TF3	15	2EP3	15	3UK3	473
1SRK	10	2EE8	76	2EOR	15	2RSJ	12	1TF6	107	2EPP	294	3ZNF	5
1TF3	45	2EL4	15	2EOU	15	2WBU	402	1TF6	137	2EPQ	385	4F6M	496
1TF3	75	2EL5	13	2EOX	15	2WBU	432	1TF6	45	2EPX	478	4F6N	496
1TF6	15	2ELR	12	2EOY	15	2YRH	13	1VA1	539	2EQ2	683	4F6N	524
1TF6	75	2ELU	12	2EP1	15	2YRK	16	1VA2	569	2GLI	172	4F6N	552
1U85	10	2ELV	12	2EP2	15	2YRM	13	1WJP	72	2GLI	233	4IS1	473
1U86	10	2ELW	12	2EPA	20	2YSO	15	1X3C	30	2GQJ	57	4ZNF	5
1UBD	298	2ELX	10	2EPA	50	2YSV	760	1YUI	36	2HGH	107		

Astral64

Astral [CHW+04] est une base de données structurale de domaines protéiques.

Nous avons défini un sous-ensemble léger d'Astral qu'on note Astral64. Pour qu'il soit relativement représentatif des différentes structures connues, nous avons aléatoirement choisi 8 domaines Astral (version 2.03 [CHW+04]) dans chacune des 4 principales classes SCOP (uniquement alpha, uniquement beta, alpha/beta, alpha+beta). Le jeu résultant contient donc 64 domaines dont les identifiants Astral sont les suivants :
d1bgab_ d1bz1a_, d1cpcl_, d1ehyb_, d1f7ca_, d1fhqa_, d1gbda_, d1gqcb_, d1lt6f_, d1s3ca_, d1uppj_, d1urpa_, d1x7sa_, d1x8mf_, d1xtvb_, d1y4vc_ d1y59t_, d2a3wn_, d2ahcb_, d2bkck_, d2c1dd_, d2c7la_, d2hbdb_, d2j73a_ d2o64a_, d2qdsa_, d2uzla_, d2vlfa_, d2xjoa_, d3az9n_, d3b2ja_, d3dcgb_ d3diea_, d3e29d_, d3eqba_, d3euya_, d3fckb_, d3hf9l_, d3hnid_, d3i5vd_ d3jxza_, d3kwaa_, d3l2yg_, d3lele_, d3m1ob_, d3m64a_, d3n6ab_, d3nbtd_ d3nhha_, d3oced_, d3qiha_, d3rdhb_, d3ruac_, d3rufa_, d3uh7b_, d3ux7c_ d3w29a_, d3zxeb_, d4actb_, d4bcqc_, d4ejja_, d4epva_, d4i83f_, d1i4td_.

Structures de capsides

Nous avons construit un jeu de données de structures de capsides afin de caractériser ce type de protéines d'une part d'un point de vue purement structural via l'identification de CF caractéristiques mais aussi dans le but de construire des HMM pour modéliser les séquences de structures conservées (utilisé dans VIRALpro par exemple).

Pour construire ce jeu, nous avons utilisé Uniprot [MC11] avec une requête cherchant les capsides et nucleocapsides (type de capside pour lequel la capside est directement reliée au matériel génétique). Nous avons ensuite appliqué un filtre taxonomique pour ne garder que les virus (et retirer les séquences codant pour des protéines se liant à des capsides par exemple), et seuls les identifiants Uniprot associés à des structures PDB ont été conservés. En réduisant la redondance (90%) des séquences des PDB, nous avons obtenu un jeu de données de 327 chaînes de capsides.

Les identifiants PDB retenus sont les suivants (sur deux pages) :

PDB	Chaîne	PDB	Chaîne	PDB	Chaîne	PDB	Chaîne	PDB	Chaîne
1A3R	H	1KQR	A	2BPA	3	2XYA	A	3S6P	C
1A3R	L	1KVP	A	2CGT	A	2YEW	A	3S6P	G
1ADU	A	1L5I	A	2CGT	O	2YEW	F	3S6P	H
1AHS	A	1LAJ	C	2CII	A	2YEW	H	3S6X	B
1AL0	B	1LVM	A	2CSE	1	2YPL	A	3S7V	C
1AUY	B	1M06	G	2CSE	T	2YPL	C	3S7X	A
1AYM	1	1M06	J	2CSE	W	2YPL	D	3TIR	A
1AYM	2	1M0F	B	2CSE	X	2YPL	E	3TS3	B
1AYM	3	1M0F	F	2EC7	A	2ZAH	C	3UAJ	A
1AYN	4	1M1C	B	2F5U	A	2ZZQ	A	3UAJ	C
1B35	A	1MEC	1	2F76	X	3CNC	E	3UAJ	D
1B35	B	1MEC	2	2F8E	A	3CNF	B	3UC0	B
1B35	C	1MEC	3	2F8E	X	3DPR	E	3UX1	A
1B35	D	1MEC	4	2FTE	E	3EOY	G	3V7O	B
1BAI	B	1MOF	A	2G8G	A	3EPF	1	3VDD	A
1BMV	2	1NCP	C	2GH8	A	3EPF	4	3VDD	B
1BVP	5	1NCP	N	2HJL	C	3EPF	R	3VDD	D
1C0M	C	1NOB	F	2HRV	B	3EXW	A	3ZED	B
1C6V	A	1NOV	C	2HWD	2	3EZK	D	3ZED	D
1C6V	X	1NOV	D	2HWE	3	3GCZ	A	3ZIF	B
1C8N	C	1P30	A	2I9F	C	3IXO	A	3ZIF	M
1CD3	4	1PGL	1	2IC6	B	3IYH	B	3ZIF	N
1CD3	G	1PGL	2	2IFO	A	3IYN	C	3ZIF	R
1D3I	I	1PHX	1	2IN2	A	3IYN	N	3ZL9	C
1D4M	1	1QIU	C	2INY	A	3IYN	O	3ZXA	C
1D4M	2	1QRJ	A	2IUN	A	3IYN	Q	4AN5	E
1DDL	B	1R1A	1	2IZW	C	3IYN	R	4AR2	A
1DNV	A	1RB8	F	2JW8	B	3IYS	E	4ATZ	A
1DWN	B	1RER	B	2KPZ	A	3IZ3	E	4ATZ	D
1DYL	A	1RHI	1	2LZP	A	3IZO	D	4BCU	A
1EAH	2	1RHI	2	2LZQ	A	3IZO	G	4BLO	L
1EAH	3	1RHI	3	2M5S	A	3IZX	A	4BP4	c
1ED1	A	1RHI	4	2M6X	A	3IZX	C	4BTQ	B
1EJ6	A	1RMV	A	2M7Y	A	3J1A	G	4CSF	k
1EV1	1	1RUE	3	2M9U	A	3J26	E	4CWU	O
1EV1	2	1RVF	H	2MKB	A	3J26	N	4CWU	T
1EV1	3	1RVF	L	2MQV	A	3J2J	B	4CWU	U
1EV1	4	1SLQ	A	2OBE	A	3J2W	D	4CWU	X
1F8V	A	1SSK	A	2OXT	D	3J2W	E	4DGB	A
1F8V	D	1STM	E	2PX5	A	3J2W	O	4DOX	B
1F8V	E	1TGE	A	2QVJ	D	3J2W	T	4FTE	B
1FPN	3	1TMF	1	2R5I	M	3J31	F	4FY8	Q
1FR5	B	1TMF	2	2R5J	D	3J31	P	4FYA	u
1FRS	C	1TMF	3	2R5K	A	3J31	Q	4GHA	C
1FZM	B	1TMF	4	2R69	H	3J3I	A	4GU4	A
1G31	A	1UF2	B	2R69	L	3J6R	C	4GVE	A
1GAV	A	1UF2	K	2TBV	C	3KEE	D	4IJS	D
1GFF	1	1UF2	S	2V33	B	3KF2	B	4J1J	C
1GFF	2	1V1H	E	2V6J	A	3KF2	D	4JZ2	A

PDB	Chaîne	PDB	Chaîne	PDB	Chaîne	PDB	Chaîne	PDB	Chaîne
1GFF	3	1VTM	P	2VRS	C	3KK5	F	4K50	A
1H7Z	B	1W8X	B	2VTU	L	3KKS	B	4LLF	M
1HB5	B	1W8X	M	2VTW	E	3L29	B	4MG3	B
1HGZ	A	1W8X	N	2W0C	F	3L89	L	4MH8	A
1HHJ	D	1W8X	P	2W0C	L	3L89	X	4MJ0	E
1HHJ	E	1X33	C	2W0C	P	3MMG	A	4NIA	D
1IFK	A	1XR5	A	2W0C	S	3NAP	A	4NYZ	A
1IFL	A	1YQ8	A	2W0C	T	3NAP	B	4NZG	C
1IFP	A	1Z1U	E	2WA1	A	3NAP	C	4O6B	B
1IHM	B	1ZA7	B	2WA1	B	3OV9	A	4O6H	D
1JMU	G	1ZN5	A	2WAZ	X	3P0S	A	4OR8	B
1JS9	C	2AEN	A	2WBV	D	3P4N	A	4POQ	J
1JVR	A	2B7F	F	2WLV	A	3PV6	A	4SBV	C
1K4R	C	2B7F	J	2WST	C	3PV6	B	4UON	A
1K5M	A	2BBV	C	2WSU	C	3QSQ	A		
1K5M	B	2BFU	L	2WV9	A	3RNV	A		
1KAC	B	2BFU	S	2XS8	A	3RQV	a		

VIRALpro

Afin de réaliser l'apprentissage des deux SVM de VIRALpro (détection de protéines de capside et détection de protéines de queue), nous avons construit respectivement deux jeux d'apprentissage.

Jeu d'apprentissage de CAPSIDpro : $\mathbf{Cpro_{train}}$

On note détaille ici le jeu utilisé pour la détection de protéines de capside. Les séquences de capside proviennent de la requête suivante sur la base de données protéique du NCBI : *capsid[Title] AND viruses[Organism] NOT putative NOT scaffolding NOT assembly NOT partial NOT precursor NOT probable NOT polyprotein.* Les 21202 séquences résultantes ont été réduites à 2648 séquences (seuil à 90% d'identité). Nous avons ajouté 483 séquences de nucléocapsides par le même processus. Finalement, on obtient 3131 séquences positives (capsides et ncléocapsides).

Pour construire la partie négative du jeu d'apprentissage, nous avons en partie des séquences aléatoirement choisies dans le jeu négatif d'apprentissage du réseau de neurones de iVireons Strctural, auquel nous avons ajouté des séquences aléatoirement choisies dans des taxonomies différentes de celle des virus dans la base de données de la GenBank. Au final, on obtient 3828 séquences négatives.

Au final, nous avons joint ce jeu d'apprentissage à celui de iVireons MCP1 :1 — noté $\mathrm{iV_{capsid}}$ —, pour obtenir 3888 séquences positives et 4650 séquences negatives. On notera ce jeu complet $\mathrm{Cpro_{train}}$.

Jeu d'apprentissage de TAILpro : $\mathbf{Tpro_{train}}$

Nous avons utilisé exactement le même processus pour la construction du jeu de données pour la détection de protéines de queue. La requête *tail[Title] AND viruses[Organism]*

NOT putative NOT scaffolding NOT assembly NOT partial NOT precursor NOT probable NOT polyprotein NOT chaperone NOT tape NOT sheath NOT baseplate NOT minor NOT assembly NOT "and" sur la base protéique du NCBI fournit 4895 séquences que nous avons réduit à 1719 séquences à 90% de non-redondance. Nous avons utilisé les mêmes 3828 séquences négatives sélectionnées pour $\text{Cpro}_{\text{train}}$. Nous avons également joint à ce jeu les séquences d'apprentissage de iVireons Tail 1 :1.

Au final, on obtient 2574 séquences positives et 4683 séquences négatives. On note ce jeu complet $\text{Tpro}_{\text{train}}$.

Validation croisée et jeu de validation

Chacun des jeux d'apprentissage est découpé aléatoirement en 10 sous-parties équilibrées en ratio positif-négatif sur lesquelles nous avons réalisé la validation croisée.

Par ailleurs, nous appelons *jeu de validation* le jeu constitué de cette dernière sous-partie (10%"du total des séquences), et lorsque nous utilisons VIRALpro sur ce jeu de validation, cela signifie qu'il a été entraîné seulement sur les 9 autres sous-parties (90% des séquences).Les identifiants des séquences utilisés pour le jeu de validation sont listés dans les fichiers *CAPSIDproValidationSet.csv* et *TAILproValidationSet.csv* disponibles sur `http://www.irisa.fr/dyliss/public/VIRALpro/SupplMat/`.

Jeu d'apprentissage et de test de iVireons

Les deux jeux de données $\text{iV}_{\text{capsid}}$ et iV_{test} sont détaillés dans les annexes de [SAA$^+$12]. Nous avons reconstitué ces jeux de données au plus proches des indications données (*e.g.* les identifiants exacts pour le réseau de neurones MCP 1 :1 n'étaient pas fournis). Les identifiants correspondant respectivement à $\text{iV}_{\text{capsid}}$ et iV_{test} sont listés dans *ivtrain.txt* et *ivtest.txt* ; ils correspondent au jeu d'apprentissage et de test du réseau de neurone dénoté MCP 1 :1 dans [SAA$^+$12] (réseau détectant les protéines majeurs de capsides (Major Capsid protein MCP) entraîné avec un ratio de 1 :1 de séquences positives vs. négatives).

Bien que chaque identifiant de séquence dans $\text{iV}_{\text{capsid}}$ est différent de iV_{test} , les séquences de ce dernier sont fortement homologues à celles du premier : 55 des 59 séquences de capsides dans iV_{test} ont un homologue dans $\text{iV}_{\text{capsid}}$ (ayant une e-value plus petite que 10^{-9}). Par ailleurs, il faut noter que $\text{iV}_{\text{capsid}}$ est très redondant : en limitant l'identité de séquence à 90% on réduit le jeu de plus de la moitié. Il n'est donc pas surprenant que la validation croisée de VIRALpro sur $\text{iV}_{\text{capsid}}$ soit meilleure que sur $\text{Cpro}_{\text{train}}$, qui est non-redondant à 90%.

Enfin, beaucoup de séquences utilisées dans iV_{test} ont été supprimées de la base de données du NCBI, et quelques unes d'entre elles ont vu leur annotation mise à jour de manière contradictoire (par exemple des séquences anciennement annotées comme queue sont désormais annotées comme protéine de capside).

C'est pourquoi nous avons utilisé iV_{test} uniquement dans le but de comparer VIRALpro à iVireons sur le même jeu d'apprentissage $\text{iV}_{\text{capsid}}$.

Jeux de données réelles à annoter

Nous avons illustré la détection de protéines de capsides et de queue avec VIRALpro sur trois jeux de données viraux précédemment publiés : sur une étude des virus à ARN sur des échantillons d'eau de mer en Colombie Britanique (dénoté RnaCoastal dans ce document), sur des séquences provenant de phages de la mer Baltique (dénoté Oresund) et enfin sur un troisième jeu de séquences provenant de phages marins séquencé par le projet "Marine Phage, Virus, and Virome Sequencing" du Broad Institute (dénoté Moore). Nous donnons ici quelques détails de la construction de ces jeux de test à partir des données publiques disponibles.

Il est intéressant de remarquer que les jeux (détaillés dans les prochains paragraphes) RnaCoastal et Oresund-Struct n'ont aucune séquence ayant une similarité significative avec notre jeu positif d'apprentissage. Par ailleurs, Oresund-Hypo, Oresund-NonStruct et MooreMarinePhages ont respectivement 0.2%, 1.3% and, 1.8% de leur séquence qui possède une homologie significative avec une séquence de notre jeu positif d'apprentissage (e-value ≤ 0.001).

RNACoastal A partir des données publiques des études viral métagénomiques publiées sur `http://virome.dbi.udel.edu` [WBP+12], nous avons choisi un jeu de données [CLS06] pour lequel il y avait un nombre raisonnable d'ORFans (*i.e.* Open Reading Frames sans homologues connus) : il s'agit d'une étude métagénomique portant sur les virus à ARN provenant de deux stations marines sur le littoral de Colombie Britanique. Sur la base Virome, seule les données de la station JP sont disponibles. On note RnaCoastal ce jeu de 86 ORFans.

Oresund Ce jeu de données provient de l'étude des génomes de 31 phages marins identifiés dans le détroit d'Oresund. Grâce à des techniques de protéomiques par spectrométrie de masse, les auteurs ont pu identifier 192 séquences comme étant des protéines strcturales de ces phages. 83% d'entre elles n'ont aucune homologie connue (e-value ≤ 0.001). Dans les données disponibles en ligne, 141 séquences avaient été identifiées par spectrométrie de masse et avaient été annotées comme "structurales". Parmi ces 141 séquences, 84 (60%) d'entre elles ne possèdent aucun *hit* dans les bases de données suivantes parmi les séquences protéiques du NCBI, ainsi que dans la base des séquences de domaines conservées CDD (NCBI conserved domain database) ni dans la base de données Pfam. On dénote par Oresund-Struct l'ensemble de ces 84 séquences.

De plus, l'étude annote comme "protéines hypothétiques" 791 séquences qui ne possèdent aucune homologie au moment de l'étude. Nous avons cependant trouvé que 10 d'entre elles possédaient une similarité significative avec des séquences de $\text{Cpro}_{\text{train}}$ (e-value ≤ 0.001). On note par Oresund-Hypo ces 791 séquences.

Enfin, 156 protéines n'ont pas été détectées par spectrométrie de masse et qui ont de plus une similarité avec des séquences non-structurales (*i.e.* un hit dans une des 3 bases de données NCBI protéique, CDD, Pfam). On note Oresund-NonStruct cet ensemble de séquences, permettant d'évaluer la spécificité de VIRALpro. Il reste à noter que parmi ces séquences, il se peut que quelques-unes d'entre elles soient tout de même des

protéines structurales. En effet, une communication avec les auteurs nous a informé que les petites séquences protéiques structurales ne sont pas visible par spectrométrie de masse et donc quelques une d'entre elles ont pu échapper à l'identification.

Moore Le projet "Marine Phage, Virus, and Virome Sequencing Project" de la fondation Gordon and Betty Moore a permis la publicatioin de génomes de phages marins dans la base de données iMicrobe `http://www.imicrobe.us`. Le code associé à ce projet dans iMicrobe est CAM_PROJ_BroadPhageGenomes. Il contient 20343 séquences protéiques parmis lesquelles 1172 n'ont aucune annotation. On dénote par Moore le jeu de données constitué de cet ensemble de 1172 séquences.

Bibliographie

[ABW04] Rolf Apweiler, Amos Bairoch, and Cathy H Wu. Protein sequence databases. *Current Opinion in Chemical Biology*, 8(1) :76 – 80, 2004.

[AGM+90] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3) :403–410, 1990.

[AHC+14] Antonina Andreeva, Dave Howorth, Cyrus Chothia, Eugene Kulesha, and Alexey G. Murzin. SCOP2 prototype : a new approach to protein structure mining. *Nucleic Acids Research*, 42(D1) :D310–D314, 2014.

[AL09] Jure Piskur Göran Lindblom Poul Nissen Morten Kjeldgaard Anders Liljas, Lars Liljas. *Textbook of Structural Biology*. World Scientific, 2009.

[AMDY11] Rumen Andonov, Noël Malod-Dognin, and Nicola Yanev. Maximum contact map overlap revisited. *Journal of Computational Biology*, 18(1) :27–41, 2011.

[AMS+97] Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast : a new generation of protein database search programs. *Nucleic acids research*, 25(17) :3389–3402, 1997.

[Anf72] Christian B Anfinsen. Studies on the principles that govern the folding of protein chains, 1972.

[BBC+02] Alex Bateman, Ewan Birney, Lorenzo Cerruti, Richard Durbin, Laurence Etwiller, Sean R. Eddy, Sam Griffiths-Jones, Kevin L. Howe, Mhairi Marshall, and Erik L. L. Sonnhammer. The pfam protein families database. *Nucleic Acids Research*, 30(1) :276–280, 2002.

[BCC+14] Martine Boccara, Mathilde Carpentier, Jacques Chomilier, François Coste, Clovis Galiez, Joël Pothier, and Alaguraj Veluchamy. Identifying distant homologous viral sequences in metagenomes using protein structure information. In *ECCB'14 Workshop on Recent Computational Advances in Metagenomics*, Strasbourg, France, September 2014.

[BENS+11] Arren Bar-Even, Elad Noor, Yonatan Savir, Wolfram Liebermeister, Dan Davidi, Dan S. Tawfik, and Ron Milo. The moderately efficient enzyme : Evolutionary and physicochemical trends shaping enzyme parameters. *Biochemistry*, 50(21) :4402–4410, 2011. PMID : 21506553.

[BKW+77] F. C. Bernstein, T. F. Koetzle, G. J. Williams, E. F. Meyer, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi. The Protein Data Bank : a computer-based archival file for macromolecular structures. *Journal of molecular biology*, 112(3) :535–542, May 1977.

[BRS04] Anirban Bhaduri, R Ravishankar, and R Sowdhamini. Conserved spatially interacting motifs of protein superfamilies : application to fold recognition and function annotation of genome data. *Proteins : Structure, Function, and Bioinformatics*, 54(4) :657–670, 2004.

[BTNK10] Inbal Budowski-Tal, Yuval Nov, and Rachel Kolodny. Fragbag, an accurate representation of protein structure, retrieves structural neighbors from the entire pdb quickly and accurately. *Proceedings of the National Academy of Sciences*, 107(8) :3481–3486, 2010.

[CBP05] Mathilde Carpentier, Sophie Brouillet, and Joël Pothier. Yakusa : A fast structural database scanning method. *Proteins : Structure, Function, and Bioinformatics*, 61(1) :137–151, 2005.

[CCA+09] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. Blast+ : architecture and applications. *BMC bioinformatics*, 10(1) :421, 2009.

[CGKG07] Yih-Shien Chiang, Tatiana I. Gelfand, Alexander E. Kister, and Israel M. Gelfand. New classification of supersecondary structures of sandwich-like proteins uncovers strict patterns of strand assemblage. *Proteins : Structure, Function, and Bioinformatics*, 68(4) :915–921, 2007.

[CGT04] A.C Camproux, R Gautier, and P Tufféry. A hidden Markov model derived structural alphabet for proteins. *Journal of Molecular Biology*, 339(3) :591–605, 2004.

[CHW+04] John-Marc M. Chandonia, Gary Hon, Nigel S. Walker, Loredana Lo Conte, Patrice Koehl, Michael Levitt, and Steven E. Brenner. The ASTRAL Compendium in 2004. *Nucleic Acids Research*, 32(Database issue) :D189–D192, January 2004.

[CLS06] Alexander I. Culley, Andrew S. Lang, and Curtis A. Suttle. Metagenomic Analysis of Coastal RNA Virus Communities. *Science*, 312(5781) :1795–1798, June 2006.

[CPZ97] Paolo Ciaccia, Marco Patella, and Pavel Zezula. M-tree : An efficient access method for similarity search in metric spaces. In *Proceedings of the 23rd*

International Conference on Very Large Data Bases, VLDB '97, pages 426–435, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[CS96] Su Yun Chung and S Subbiah. A structural explanation for the twilight zone of protein sequence homology. *Structure*, 4(10) :1123 – 1127, 1996.

[CST00] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[CT65] James Cooley and John Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90) :297–301, 1965.

[DB79] David L. Davies and Donald W. Bouldin. A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1(2) :224–227, February 1979.

[dBEH00a] A.G. de Brevern, C. Etchebest, and S. Hazout. Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. *Proteins : Structure, Function, and Bioinformatics*, 41(3) :271–287, 2000.

[DBEH00b] AG De Brevern, Catherine Etchebest, and Serge Hazout. Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. *Proteins : Structure, Function, and Bioinformatics*, 41(3) :271–287, 2000.

[DC12] Linda Dib and Alessandra Carbone. Protein fragments : Functional and structural roles of their coevolution networks. *PLoS ONE*, 2012.

[DKL+14] James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M. Deane. Sabdab : the structural antibody database. *Nucleic Acids Research*, 42(D1) :D1140–D1146, 2014.

[dLNB12] Pietro di Lena, Ken Nagata, and Pierre Baldi. Deep architectures for protein contact map prediction. *Bioinformatics*, 28(19) :2449–2457, 2012.

[DOM+08] A. K. Dunker, C. J. Oldfield, J. Meng, P. Romero, J. Y. Yang, J. W. Chen, V. Vacic, Z. Obradovic, and V. N. Uversky. The unfoldomics decade : an update on intrinsically disordered proteins. *BMC Genomics*, 9(Suppl 2) :S1+, September 2008.

[Dra99] John W Drake. The distribution of rates of spontaneous mutation over viruses, prokaryotes, and eukaryotes. *Annals of the New York Academy of Sciences*, 870(1) :100–107, 1999.

[DSS+10] Jose M Duarte, Rajagopal Sathyapriya, Henning Stehr, Ioannis Filippis, and Michael Lappe. Optimal contact definition for reconstruction of contact maps. *BMC bioinformatics*, 11(1) :283, 2010.

[EBHdB05] Catherine Etchebest, Cristina Benros, Serge Hazout, and Alexandre G. de Brevern. A structural alphabet for local protein structures : Improved prediction methods. *Proteins : Structure, Function, and Bioinformatics*, 59(4) :810–827, 2005.

[Edd98] S R Eddy. Profile hidden markov models. *Bioinformatics*, 14(9) :755–763, 1998.

[Edg04] Robert C Edgar. Muscle : multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research*, 32(5) :1792–1797, 2004.

[Eli03] Isaac Elias. Settling the intractability of multiple alignment. In Toshihide Ibaraki, Naoki Katoh, and Hirotaka Ono, editors, *ISAAC*, volume 2906 of *Lecture Notes in Computer Science*, pages 352–363. Springer, 2003.

[ER05] Robert A Edwards and Forest Rohwer. Viral metagenomics. *Nature Reviews Microbiology*, 3(6) :504–510, 2005.

[FISS03] Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *J. Mach. Learn. Res.*, 4 :933–969, December 2003.

[GC15] Clovis Galiez and François Coste. Amplitude spectrum distance : measuring the global shape divergence of protein fragments. *BMC bioinformatics*, 16(1) :256, 2015.

[GH02] Richard A. George and Jaap Heringa. An analysis of protein domain linkers : their classification and role in protein folding. *Protein Engineering*, 15(11) :871–879, 2002.

[GKK+11] Martin Gebser, Benjamin Kaufmann, Roland Kaminski, Max Ostrowski, Torsten Schaub, and Marius Schneider. Potassco : The potsdam answer set solving collection. *Ai Communications*, 24(2) :107–124, 2011.

[GMCB15] Clovis Galiez, Christophe Magnan, François Coste, and Pierre Baldi. Viralpro : a new suite for identifying viral capsid and tail sequences. *Submitted*, 2015.

[Gor04] G Gordon. Support vector machines and kernel methods. 2004.

[GT10] Frédéric Guyon and Pierre Tufféry. Assessing 3D scores for protein structure fragment mining. *Open Access Bioinformatics*, 2 :67–77, 2010.

[GT13] Frédéric Guyon and Pierre Tufféry. Fast protein fragment similarity scoring using a binet–cauchy kernel. *Bioinformatics*, 2013.

[HH92] Steven Henikoff and Jorja G Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22) :10915–10919, 1992.

[HKO04] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent component analysis*, volume 46. John Wiley & Sons, 2004.

[Hol11] Edward C Holmes. The evolution of endogenous viral elements. *Cell host & microbe*, 10(4) :368–377, 2011.

[HP00] Liisa Holm and Jong Park. Dalilite workbench for protein structure comparison. *Bioinformatics*, 16(6) :566–567, 2000.

[HRLR09] Najeeb Halabi, Olivier Rivoire, Stanislas Leibler, and Rama Ranganathan. Protein sectors : evolutionary units of three-dimensional structure. *Cell*, 138(4) :774–786, 2009.

[HS13] Bonnie L. Hurwitz and Matthew B. Sullivan. The pacific ocean virome (pov) : A marine viral metagenomic dataset and associated protein clusters for quantitative viral ecology. *PLoS ONE*, 8(2) :e57355, 02 2013.

[HSS+13] Karin Holmfeldt, Natalie Solonenko, Manesh Shah, Kristen Corrier, Lasse Riemann, Nathan C. VerBerkmoes, and Matthew B. Sullivan. Twelve previously unknown phage genera are ubiquitous in global oceans. *Proceedings of the National Academy of Sciences*, 110(31) :12798–12803, 2013.

[Jai89] Anil K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[JBCP12] David T. Jones, Daniel W. A. Buchan, Domenico Cozzetto, and Massimiliano Pontil. Psicov : precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, 28(2) :184–190, 2012.

[JECT02] Inge Jonassen, Ingvar Eidhammer, Darrell Conklin, and William R. Taylor. Structure motif discovery and mining the pdb. *Bioinformatics*, 18(2) :362–367, 2002.

[Jon97] Inge Jonassen. Efficient discovery of conserved patterns using a pattern graph. *Computer applications in the biosciences : CABIOS*, 13(5) :509–522, 1997.

[Jus15] Maud Jusot. Caractérisation en séquence et en structure des protéines virales. *Rapport de Master 2 encadré par F. Coste et M. Carpentier*, 2015.

[KA90] Samuel Karlin and Stephen F Altschul. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proceedings of the National Academy of Sciences*, 87(6) :2264–2268, 1990.

[Kab76] W. Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5) :922–923, September 1976.

[KB11] Mart Krupovic and Dennis H Bamford. Double-stranded {DNA} viruses : 20 families and only five different architectural principles for virion assembly. *Current Opinion in Virology*, 1(2) :118 – 124, 2011. Virus structure and function.

[KBD+58] John C Kendrew, G Bodo, Howard M Dintzis, RG Parrish, Harold Wyckoff, and David C Phillips. A three-dimensional model of the myoglobin molecule obtained by x-ray analysis. *Nature*, 181(4610) :662–666, 1958.

[KMKM02] Kazutaka Katoh, Kazuharu Misawa, Kei-ichi Kuma, and Takashi Miyata. Mafft : a novel method for rapid multiple sequence alignment based on fast fourier transform. *Nucleic acids research*, 30(14) :3059–3066, 2002.

[Koe01] Patrice Koehl. Protein structure similarities. *Current Opinion in Structural Biology*, 11(3) :348–353, June 2001.

[LBB+07] M.A. Larkin, G. Blackshields, N.P. Brown, R. Chenna, P.A. McGettigan, H. McWilliam, F. Valentin, I.M. Wallace, A. Wilm, R. Lopez, J.D. Thompson, T.J. Gibson, and D.G. Higgins. Clustal w and clustal x version 2.0. *Bioinformatics*, 23(21) :2947–2948, 2007.

[LBG+08] Shuai C. Li, Dongbo Bu, Xin Gao, Jinbo Xu, and Ming Li. Designing succinct structural alphabets. *Bioinformatics*, 24(13) :i182–189, July 2008.

[LCAH+00] Loredana Lo Conte, Bart Ailey, Tim J. P. Hubbard, Steven E. Brenner, Alexey G. Murzin, and Cyrus Chothia. Scop : a structural classification of proteins database. *Nucleic Acids Research*, 28(1) :257–259, 2000.

[LCWI01] Giuseppe Lancia, Robert Carr, Brian Walenz, and Sorin Istrail. 101 optimal pdb structure alignments : A branch-and-cut algorithm for the maximum contact map overlap problem. *Proceedings of the Fifth Annual International Conference on Computational Biology*, pages 193–202, 2001.

[Lev76] Michael Levitt. A simplified representation of protein conformations for rapid simulation of protein folding. *Journal of molecular biology*, 104(1) :59–107, 1976.

[LKS+14] Yoav Lehahn, Ilan Koren, Daniella Schatz, Miguel Frada, Uri Sheyn, Emmanuel Boss, Shai Efrati, Yinon Rudich, Miri Trainic, Shlomit Sharoni, Christian Laber, Giacomo R. DiTullio, Marco J.L. Coolen, Ana Maria Martins, Benjamin A.S. Van Mooy, Kay D. Bidle, and Assaf Vardi. Decoupling physical from biological processes to assess the impact of viruses on a mesoscale algal bloom. *Current Biology*, 24(17) :2041 – 2046, 2014.

[LPK09] Quan Le, Gianluca Pollastri, and Patrice Koehl. Structural Alphabets for Protein Structure Classification : A Comparison Study. *Journal of Molecular Biology*, 387(2) :431–450, March 2009.

[Mac78] Saunders MacLane. *Categories for the working mathematician*, volume 5. Springer Science - Business Media, 1978.

[MB14] Christophe N. Magnan and Pierre Baldi. SSpro/ACCpro 5 : almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, 30(18) :2592–2597, September 2014.

[MC11] Michele Magrane and UniProt Consortium. Uniprot knowledgebase : a hub of integrated protein data. *Database*, 2011, 2011.

[MCD+14] Alex Mitchell, Hsin-Yu Chang, Louise Daugherty, Matthew Fraser, Sarah Hunter, Rodrigo Lopez, Craig McAnulla, Conor McMenamin, Gift Nuka, Sebastien Pesseat, Amaia Sangrador-Vegas, Maxim Scheremetjew, Claudia Rato, Siew-Yit Yong, Alex Bateman, Marco Punta, Teresa K. Attwood, Christian J.A. Sigrist, Nicole Redaschi, Catherine Rivoire, Ioannis Xenarios, Daniel Kahn, Dominique Guyot, Peer Bork, Ivica Letunic, Julian Gough, Matt Oates, Daniel Haft, Hongzhan Huang, Darren A. Natale, Cathy H. Wu, Christine Orengo, Ian Sillitoe, Huaiyu Mi, Paul D. Thomas, and Robert D. Finn. The interpro protein families database : the classification resource after 15 years. *Nucleic Acids Research*, 2014.

[MCS+11] Debora S. Marks, Lucy J. Colwell, Robert Sheridan, Thomas A. Hopf, Andrea Pagnani, Riccardo Zecchina, and Chris Sander. Protein 3D structure computed from evolutionary sequence variation. *PloS one*, 6(12) :e28766+, December 2011.

[MDF+14] Bohdan Monastyrskyy, Daniel D'Andrea, Krzysztof Fidelis, Anna Tramontano, and Andriy Kryshtafovych. Evaluation of residue–residue contact prediction in casp10. *Proteins : Structure, Function, and Bioinformatics*, 82 :138–153, 2014.

[MG99] K S Makarova and N V Grishin. Thermolysin and mitochondrial processing peptidase : how far structure-functional convergence goes. *Protein Science*, 8(11) :2537–40, 1999.

[MHS12] Debora S Marks, Thomas A Hopf, and Chris Sander. Protein structure prediction from sequence variation. *Nature biotechnology*, 30(11) :1072–1080, 2012.

[MPZG02] Antoine Marin, Joël Pothier, Karel Zimmermann, and Jean-François Gibrat. Frost : A filter-based fold recognition method. *Proteins : Structure, Function, and Bioinformatics*, 49(4) :493–509, 2002.

[MSC13] Shintaro Minami, Kengo Sawada, and George Chikenji. Mican : a protein structure alignment algorithm that can handle multiple-chains, inverse alignments, calpha only models, alternative alignments, and non-sequential alignments. *BMC Bioinformatics*, 14(1) :24, 2013.

[NHH00] Cédric Notredame, Desmond G Higgins, and Jaap Heringa. T-coffee : a novel method for fast and accurate multiple sequence alignment1. *Journal of Molecular Biology*, 302(1) :205 – 217, 2000.

[NLJ11] Benjamin North, Andreas Lehmann, and Roland L. Dunbrack Jr. A new clustering of antibody {CDR} loop conformations. *Journal of Molecular Biology*, 406(2) :228 – 256, 2011.

[NW70] Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3) :443 – 453, 1970.

[PMBB00] Anna R Panchenko, Aron Marchler-Bauer, and Stephen H Bryant. Combination of threading potentials and sequence profiles improves fold recognition. *Journal of Molecular Biology*, 296(5) :1319 – 1331, 2000.

[PNP+15] Stéphane Pesant, Fabrice Not, Marc Picheral, Stefanie Kandels-Lewis, Noan Le Bescot, Gabriel Gorsky, Daniele Iudicone, Eric Karsenti, Sabrina Speich, Romain Troublé, et al. Open science resources for the discovery and analysis of tara oceans data. *Scientific Data*, 2, 2015.

[PO08] Marco Punta and Yanay Ofran. The rough guide to in silico function prediction, or how to use sequence and structure information to predict protein function. *PLoS Comput Biol*, 4(10) :e1000160, 2008.

[PR05] Marco Punta and Burkhard Rost. Profcon : novel prediction of long-range contacts. *Bioinformatics*, 21(13) :2960–2968, 2005.

[PSSC08] Ganesan Pugalenthi, Ponnuthurai N Suganthan, Ramanathan Sowdhamini, and Saikat Chakrabarti. Megamotifbase : a database of structural motifs in protein families and superfamilies. *Nucleic acids research*, 36(suppl 1) :D218–D221, 2008.

[PTBM12] Kim D. Pruitt, Tatiana Tatusova, Garth R. Brown, and Donna R. Maglott. Ncbi reference sequences (refseq) : current status, new features and genome annotation policy. *Nucleic Acids Research*, 40(D1) :D130–D135, 2012.

[RF03] Peter Røgen and Boris Fain. Automatic classification of protein structure by using Gauss integrals. *Proceedings of the National Academy of Sciences, Vol. 100, No. 1.*, pages 119–124, 2003.

[Rig48] Jacques Riguet. Relations binaires, fermetures, correspondances de galois. *Bulletin de la société mathématique de France*, 76 :114–155, 1948.

[RKH11] Kamisetty Ramamohan Rao, Do Nyeon Kim, and Jae Jeong Hwang. *Fast Fourier Transform-Algorithms and Applications*. Springer Science & Business Media, 2011.

[Ros97] Burkhard Rost. Protein structures sustain evolutionary drift. *Folding and Design*, 2, Supplement 1 :S19 – S24, 1997.

[RRAS08] Loïc Royer, Matthias Reimann, Bill Andreopoulos, and Michael Schroeder. Unraveling protein networks with power graph analysis. *PLoS Computational Biology*, 4(7), 2008.

[SAA+12] Victor Seguritan, Nelson Alves, Michael Arnoult, Amy Raymond, Don Lorimer, Alex B. Burgin, Peter Salamon, and Anca M. Segall. Artificial Neural Networks Trained to Detect Viral and Phage Structural Proteins. *PLoS Comput Biol*, 8(8) :e1002657+, August 2012.

[SB04] J. Shapiro and D. Brutlag. FoldMiner : structural motif discovery using an improved superposition algorithm. *Protein Sci*, 13(1) :278–294, January 2004.

[SCC+13a] Christian J. A. Sigrist, Edouard De Castro, Lorenzo Cerutti, Béatrice A. Cuche, Nicolas Hulo, Alan Bridge, Lydie Bougueleret, and Ioannis Xenarios. New and continuing developments at prosite. *Nucleic Acids Research*, 41(Database-Issue) :344–347, 2013.

[SCC+13b] Christian J. A. Sigrist, Edouard De Castro, Lorenzo Cerutti, Béatrice A. Cuche, Nicolas Hulo, Alan Bridge, Lydie Bougueleret, and Ioannis Xenarios. New and continuing developments at prosite. *Nucleic Acids Research*, 41(Database-Issue) :344–347, 2013.

[SKHB97a] Kim T. Simons, Charles Kooperberg, Enoch Huang, and David Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions. *Journal of Molecular Biology*, 268(1) :209–225, April 1997.

[SKHB97b] Kim T. Simons, Charles Kooperberg, Enoch Huang, and David Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions1. *Journal of Molecular Biology*, 268(1) :209 – 225, 1997.

[SLC+15] Ian Sillitoe, Tony E. Lewis, Alison Cuff, Sayoni Das, Paul Ashford, Natalie L. Dawson, Nicholas Furnham, Roman A. Laskowski, David Lee, Jonathan G. Lees, Sonja Lehtinen, Romain A. Studer, Janet Thornton, and Christine A. Orengo. CATH : comprehensive structural and functional annotations for genome sequences. *Nucleic Acids Research*, 43(D1) :D376–D381, January 2015.

[Sut07] Curtis A. Suttle. Marine viruses — major players in the global ecosystem. *Nature Reviews Microbiology*, 2007.

[SW81] Temple F Smith and Michael S Waterman. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1) :195–197, 1981.

[SW10] RoyD. Sleator and Paul Walsh. An overview of in silico protein function prediction. *Archives of Microbiology*, 192(3) :151–155, 2010.

[TGS+06] Manoj Tyagi, Venkataraman S Gowri, Narayanaswamy Srinivasan, Alexandre G de Brevern, and Bernard Offmann. A substitution matrix for structural alphabet based on structural alignment of homologous proteins and its applications. *PROTEINS : Structure, Function, and Bioinformatics*, 65(1) :32–39, 2006.

[TKF91] Jeffrey L Thorne, Hirohisa Kishino, and Joseph Felsenstein. An evolutionary model for maximum likelihood alignment of dna sequences. *Journal of Molecular Evolution*, 33(2) :114–124, 1991.

[WAK12] Inken Wohlers, Rumen Andonov, and Gunnar W. Klau. Optimal DALI protein structure alignment. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, page 20, November 2012. RR-7915 RR-7915.

[WBP+12] K. Eric Wommack, Jaysheel Bhavsar, Shawn W. Polson, Jing Chen, Michael Dumas, Sharath Srinivasiah, Megan Furman, Sanchita Jamindar, and Daniel J. Nasko. VIROME : a standard operating procedure for analysis of viral metagenome sequences. *Standards in Genomic Sciences*, 6(3) :427–439, July 2012.

[WMBB00] S. J. Wheelan, A. Marchler-Bauer, and S. H. Bryant. Domain size distributions can predict domain boundaries. *Bioinformatics*, 16(7) :613–618, 2000.

[XCLM13] Cui Xuefeng, Li Shuai Cheng, He Lin, and Li Ming. Fingerprinting protein structures effectively and efficiently. *Bioinformatics*, 2013.

[ZS04] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins : Structure, Function, and Bioinformatics*, 57(4) :702–710, 2004.

[ZS05] Yang Zhang and Jeffrey Skolnick. Tm-align : A protein structure alignment algorithm based on tm-score. *Nucleic Acids Research*, 33 :2302–2309, 2005.